Supplementary Material for
"The Continuous Categorical: A Novel Simplex-Valued Exponential Family"

A. Derivation of the Normalizing Constant

For clarity, we start by recalling the expression that we aim to show (equation 7 from the main text), and we make the dependence on K explicit by writing $C_K(\boldsymbol{\eta})$:

$$C_K(\boldsymbol{\eta}) = \left((-1)^{K+1} \sum_{k=1}^{K} \frac{\exp\left(\eta_k\right)}{\prod_{i \neq k} \left(\eta_i - \eta_k\right)} \right)^{-1}. \tag{1}$$

Proof: we proceed by induction on K. The base case $K = 2$ can be integrated directly:

$$\begin{aligned} C_2(\boldsymbol{\eta}) &= \left(\int_0^1 \exp(\eta_1 x_1) dx_1 \right)^{-1} \\ &= \left(\frac{e^{\eta_1} - 1}{\eta_1} \right)^{-1} \\ &= \left(-\frac{e^{\eta_1}}{\eta_2 - \eta_1} - \frac{e^{\eta_2}}{\eta_1 - \eta_2} \right)^{-1}, \end{aligned} \tag{2}$$

where the last equality follows from $\eta_2 = 0$.

For the inductive step, we assume that equation 1 gives the correct normalizing constant for $K - 1$, and compute the integral for K:

$$\begin{aligned} C_K(\boldsymbol{\eta})^{-1} &= \int_{\mathbb{S}^{K-1}} \exp(\boldsymbol{\eta}^\top \mathbf{x}) d\mu \\ &= \int_0^1 \int_0^{1-x_1} \cdots \int_0^{1-x_1-\cdots-x_{K-2}} \exp\left(\sum_{i=1}^{K-1} \eta_i x_i \right) dx_{K-1} \cdots dx_2 dx_1. \end{aligned} \tag{3}$$

For the innermost integral, we have:

$$\begin{aligned} &\int_0^{1-x_1-\cdots-x_{K-2}} \exp\left(\sum_{i=1}^{K-1} \eta_i x_i \right) dx_{K-1} \\ &\qquad = \exp\left(\sum_{i=1}^{K-2} \eta_i x_i \right) \int_0^{1-x_1-\cdots-x_{K-2}} \exp(\eta_{K-1} x_{K-1}) dx_{K-1} \\ &\qquad = \exp\left(\sum_{i=1}^{K-2} \eta_i x_i \right) \left[\frac{1}{\eta_{K-1}} \exp(\eta_{K-1} t) \right]_{t=0}^{t=1-x_1-\cdots-x_{K-2}} \\ &\qquad = \frac{1}{\eta_{K-1}} \exp\left(\sum_{i=1}^{K-2} \eta_i x_i \right) \left[\exp(\eta_{K-1}(1 - x_1 - \cdots - x_{K-2})) - 1 \right] \\ &\qquad = \frac{1}{(\eta_{K-1} - \eta_K)} \left[\exp(\eta_{K-1}) \exp\left(\sum_{i=1}^{K-2} (\eta_i - \eta_{K-1}) x_i \right) - \exp\left(\sum_{i=1}^{K-2} \eta_i x_i \right) \right]. \end{aligned} \tag{4}$$

Letting $\eta_i^{(1)} = \eta_i - \eta_{K-1}$ for $i = 1, \dots, K-1$, by inductive hypothesis we have that:

$$\begin{aligned}
C_{K-1}(\boldsymbol{\eta}^{(1)})^{-1} &= \int_0^1 \int_0^{1-x_1} \cdots \int_0^{1-x_1-\cdots-x_{K-3}} \exp\left(\sum_{i=1}^{K-2} (\eta_i - \eta_{K-1}) x_i\right) dx_{K-2} \cdots dx_2 dx_1 \\
&= (-1)^K \sum_{k=1}^{K-1} \frac{\exp\left(\eta_k^{(1)}\right)}{\prod_{i \neq k} \left(\eta_i^{(1)} - \eta_k^{(1)}\right)} \\
&= (-1)^K \sum_{k=1}^{K-1} \frac{\exp\left(\eta_i - \eta_{K-1}\right)}{\prod_{i \neq k} \left(\eta_i - \eta_k\right)}. \qquad (5)
\end{aligned}$$

Similarly, letting $\eta_i^{(2)} = \eta_i$ for $i = 1, \dots, K-2$, and $\eta_{K-1}^{(2)} = 0$, we have that:

$$\begin{aligned}
C_{K-1}(\boldsymbol{\eta}^{(2)})^{-1} &= \int_0^1 \int_0^{1-x_1} \cdots \int_0^{1-x_1-\cdots-x_{K-3}} \exp\left(\sum_{i=1}^{K-2} \eta_i x_i\right) dx_{K-2} \cdots dx_2 dx_1 \\
&= (-1)^K \sum_{k=1}^{K-1} \frac{\exp\left(\eta_k^{(2)}\right)}{\prod_{i \neq k} \left(\eta_i^{(2)} - \eta_k^{(2)}\right)} \\
&= (-1)^K \left[\sum_{k=1}^{K-2} \frac{\exp\left(\eta_k\right)}{\left(-\eta_k\right) \prod_{i \neq k} \left(\eta_i - \eta_k\right)} + \frac{1}{\prod_{i=1}^{K-2} \eta_i}\right] \\
&= (-1)^K \left[\sum_{k=1}^{K-2} \frac{\exp\left(\eta_k\right)}{-\left(\eta_k - \eta_K\right) \prod_{i \neq k} \left(\eta_i - \eta_k\right)} + \frac{\exp(\eta_K)}{\prod_{i=1}^{K-2} (\eta_i - \eta_K)}\right]. \qquad (6)
\end{aligned}$$

Plugging (5) and (6) back into (4), we find:

$$C_K(\boldsymbol{\eta})^{-1} = (-1)^{K+1} \sum_{k=1}^{K} R_k(\boldsymbol{\eta}) \exp\left(\eta_k\right),$$

where the coefficients $R_k(\boldsymbol{\eta})$ gather the terms that multiply each $\exp(\eta_k)$ term. For $k = 1, \dots, K-2$, both (5) and (6) contribute to the coefficient:

$$\begin{aligned}
R_k(\boldsymbol{\eta}) &= \frac{1}{\eta_{K-1} - \eta_K} \left[-\frac{1}{\prod_{1 \leq i \leq K-1, i \neq k} (\eta_i - \eta_k)} + \frac{1}{\prod_{1 \leq i \leq K, i \neq k, i \neq K-1} (\eta_i - \eta_k)}\right] \\
&= \frac{1}{\eta_{K-1} - \eta_K} \left[\frac{-\eta_K + \eta_k + \eta_{K-1} - \eta_k}{\prod_{1 \leq i \leq K, i \neq k} (\eta_i - \eta_k)}\right] \\
&= \frac{1}{\prod_{i \neq k} (\eta_i - \eta_k)}. \qquad (7)
\end{aligned}$$

The $(K-1)$th coefficient can be computed more easily as it only appears in (5):

$$\begin{aligned}
R_{K-1}(\boldsymbol{\eta}) &= -\frac{1}{(\eta_{K-1} - \eta_K)} \frac{1}{\prod_{1 \leq i \leq K-2} (\eta_i - \eta_{K-1})} \\
&= \frac{1}{\prod_{i \neq K-1} (\eta_i - \eta_{K-1})}, \qquad (8)
\end{aligned}$$

and similarly, the Kth coefficient appears only in (6):

$$\begin{aligned}
R_K(\boldsymbol{\eta}) &= \frac{1}{(\eta_{K-1} - \eta_K)} \frac{1}{\prod_{1 \leq i \leq K-2} (\eta_i - \eta_{K-1})} \\
&= \frac{1}{\prod_{i \neq K} (\eta_i - \eta_K)}. \qquad (9)
\end{aligned}$$

This completes the proof. □

Remark. For completeness, we also include the normalizing constant written in terms of the parameterization of the original density in equation 2 of the main text:

$$\int_{\mathbb{S}^{K-1}} \prod_{i=1}^{K} \lambda_i^{x_i} d\mu(\mathbf{x}) = (-1)^{K+1} \sum_{k=1}^{K} \frac{\lambda_k}{\prod_{i \neq k} \log \frac{\lambda_i}{\lambda_k}}.$$

B. Additional Properties of the CC Distribution

B.1. Mean and Covariance

As mentioned in the main manuscript (section 3.5), by standard properties of exponential families, the mean and covariance of the CC can be obtained by differentiating the normalizing constant. For completeness, we include these results here. If $\mathbf{x} \sim \mathcal{CC}(\boldsymbol{\eta})$, then the mean of $\mathbf{x}$ is given by:

$$\mathbb{E}[x_i] = -\frac{\partial}{\partial \eta_i} \log C(\boldsymbol{\eta}), \tag{10}$$

and the covariance is given by:

$$\mathrm{cov}(x_i, x_j) = -\frac{\partial^2}{\partial \eta_i \partial \eta_j} \log C(\boldsymbol{\eta}). \tag{11}$$

B.2. KL Divergence

The KL divergence between two CC variates can be computed directly from their means:

$$\begin{aligned} KL(p(\mathbf{x}|\boldsymbol{\eta})||p(\mathbf{x}|\tilde{\boldsymbol{\eta}})) &= \mathbb{E}_{p(\mathbf{x}|\boldsymbol{\eta})}\left[\log \frac{p(\mathbf{x}|\boldsymbol{\eta})}{p(\mathbf{x}|\tilde{\boldsymbol{\eta}})}\right] \\ &= \mathbb{E}_{p(\mathbf{x}|\boldsymbol{\eta})}\left[\log C(\boldsymbol{\eta}) - \log C(\tilde{\boldsymbol{\eta}}) + \sum_{i=1}^{K-1} (\eta_i - \tilde{\eta}_i) x_i\right] \\ &= \log C(\boldsymbol{\eta}) - \log C(\tilde{\boldsymbol{\eta}}) + (\boldsymbol{\eta} - \tilde{\boldsymbol{\eta}})^\top \mathbb{E}_{p(\mathbf{x}|\boldsymbol{\eta})}[\mathbf{x}]. \end{aligned} \tag{12}$$

B.3. Moment Generating Function

The moment generating function of the CC distribution can be written directly in terms of the normalizing constant:

$$\begin{aligned} M_{\mathbf{x}}(\mathbf{t}) &= \mathbb{E}[e^{\mathbf{t}^\top \mathbf{x}}] \\ &= \int_{\mathbb{S}^{K-1}} e^{\mathbf{t}^\top \mathbf{x}} C(\boldsymbol{\eta}) e^{\boldsymbol{\eta}^\top \mathbf{x}} d\mu \\ &= C(\boldsymbol{\eta}) \int_{\mathbb{S}^{K-1}} e^{(\mathbf{t}+\boldsymbol{\eta})^\top \mathbf{x}} d\mu \\ &= \frac{C(\boldsymbol{\eta})}{C(\mathbf{t}+\boldsymbol{\eta})}. \end{aligned} \tag{13}$$

The characteristic function can be derived similarly.

B.4. Marginalization

Unlike the Dirichlet, the CC is not preserved under marginalization, even when allowing transformations of the parameter vector. In other words, if $(x_1, \ldots, x_{K-1}) \sim \mathcal{CC}(\eta_1, \ldots, \eta_{K-1})$, then it is not true that $x_1 \sim \mathcal{CC}(\eta_1)$, nor that $(x_1, \ldots, x_{K-2}) \sim \mathcal{CC}(\eta_1, \ldots, \eta_{K-2})$. It is not even true that $x_1 \sim \mathcal{CC}(\tilde{\eta}_1)$, nor that $(x_1, \ldots, x_{K-2}) \sim \mathcal{CC}(\tilde{\eta}_1, \ldots, \tilde{\eta}_{K-1})$, for any $\tilde{\boldsymbol{\eta}}$. This can be seen easily by integrating out the case $K = 3$:

$$\begin{aligned}\int_0^{1-x_1} C(\eta_1,\eta_2)\exp(\eta_1 x_1 + \eta_2 x_2)dx_2 &= C(\eta_1,\eta_2)\exp(\eta_1 x_1)\left[\frac{\exp(\eta_2 t)}{\eta_2}\right]_{t=0}^{t=1-x_1} \\ &= \frac{C(\eta_1,\eta_2)\exp(\eta_2)}{\eta_2}\exp((\eta_1-\eta_2)x_1) - \frac{C(\eta_1,\eta_2)}{\eta_2}\exp(\eta_1 x_1), \end{aligned} \tag{14}$$

which is not of the form $C(\tilde{\eta}_1)\exp(\tilde{\eta}_1 x_1)$ for any $\tilde{\eta}_1$.

As a direct consequence, we cannot use a stick-breaking construction (Connor & Mosimann, 1969; Paisley et al., 2010) to simulate CC variates from 1-dimensional CB variates, as with the Dirichlet and the Beta distributions.

C. Sampling

In this section, we develop sampling algorithms for the CC distribution and analyze their performance empirically. We also describe how to use our samplers to obtain reparameterization gradients (Kingma & Welling, 2014).

C.1. The 'Naive' Rejection Sampler

Given the form of the CC density function, a rejection sampling scheme follows readily by combining independent 1-dimensional CB draws (algorithm 1).

Algorithm 1 Naive sampler

```
Input: target distribution $\mathcal{CC}(\boldsymbol{\lambda})$.
Output: sample $\mathbf{x}$ drawn from target.
1: For $i = 1, \ldots, K-1$, draw $x_i \sim \mathcal{CC}(\lambda_i, \lambda_K)$ independently.
2: If $\sum_{i=1}^{K-1} x_i > 1$, go back to step 1, otherwise return $\mathbf{x} = (x_1, \ldots, x_{K-1})$.
```

To see why algorithm 1 achieves the desired distribution, firstly note that by independence, the distribution produced in step 1 is:

$$p_{\text{step1}}(\mathbf{x}) \propto \prod_{i=1}^{K-1} \lambda_i^{x_i} \lambda_K^{1-x_i} \propto \lambda_1^{x_1} \cdots \lambda_{K-1}^{x_{K-1}} \lambda_K^{1-x_1-\cdots-x_{K-1}}. \tag{15}$$

This is precisely the density we seek, except it is drawn on $[0,1]^{K-1}$ instead of the simplex. Step 2 rejects all samples that fall outside the simplex, thus achieving the target distribution.

The obvious shortcoming of this sampling approach is that, even for moderate values of K, the proportion of rejections becomes large. This is particularly troublesome in the balanced case, $\mathbf{x} \sim \mathcal{CC}(1/K, \ldots, 1/K)$, which is equivalent to drawing uniformly on $[0,1]^{K-1}$ and rejecting whenever we fall outside of a simplex of measure $1/(K-1)!$. In other words, we accept with a probability that decays factorially in dimension.

C.1.1. Reparameterization

The 1-dimensional CB distribution can be reparameterized using the analytical expression for the inverse CDF, derived by Loaiza-Ganem & Cunningham (2019). In this section we extend the strategy to a multivariate analogue for the CC distribution. The underlying idea is that the rejection step in algorithm 1 only depends on the L_1 norm of the proposal, but not on the parameter. This implies that, once we find an accepted proposal, we can use the inverse CDF reparameterization directly, without requiring a correction term as per the general framework for acceptance-rejection reparameterization gradients (Naesseth et al., 2017).

Our aim is to write $\mathbf{x} = g(\mathbf{u}, \boldsymbol{\lambda})$, where the density of $\mathbf{u}$ does not depend on $\boldsymbol{\lambda}$. To this end, write $F(x|\lambda_i, \lambda_K)$ for the CDF of $x \sim \mathcal{CC}(\lambda_i, \lambda_K)$. Note that this expression will follow readily from an equivalent CB distribution, as $\mathcal{CC}(\lambda_i, \lambda_K) = \mathcal{CB}(\lambda_i/(\lambda_i + \lambda_K))$. For each $i = 1, \ldots, K-1$, applying the inverse CDF component-wise on each of $u_i \overset{iid}{\sim} U(0,1)$ results in $F^{-1}(u_i|\lambda_i, \lambda_K) \sim \mathcal{CC}(\lambda_i, \lambda_K)$. Thus, the vector

$$\mathbf{F}^{-1}(\mathbf{u}|\boldsymbol{\lambda}) := [F^{-1}(u_1|\lambda_1, \lambda_K), \ldots, F^{-1}(u_{K-1}|\lambda_{K-1}, \lambda_K)] \tag{16}$$

provides a differentiable reparameterization of the distribution $\mathcal{CC}(\boldsymbol{\lambda})$, provided $\mathbf{u}$ was drawn from the pre-image of the simplex $\mathbb{S}^{K-1}$ under the mapping $\mathbf{F}^{-1}$, or in other words, provided that $\mathbf{u} \in \mathbf{F}(\mathbb{S}^{K-1})$. The rejection step simply guarantees that we find a sample of uniforms inside this region, but once we have found such a sample, it will lie in the interior of the region with probability 1, and therefore we can differentiate through the transformation as desired:

$$\frac{\partial \mathbf{x}}{\partial \boldsymbol{\lambda}} = \frac{\partial}{\partial \boldsymbol{\lambda}} \mathbf{F}^{-1}(\mathbf{u}|\boldsymbol{\lambda}). \tag{17}$$

We formalize this reparameterization in algorithm 2.

Algorithm 2 Reparameterized rejection sampler

```
Input: target distribution $\mathcal{CC}(\boldsymbol{\lambda})$.
Output: a sample $\mathbf{u}$ such that $\mathbf{F}^{-1}(\mathbf{u}|\boldsymbol{\lambda}) \sim \mathcal{CC}(\boldsymbol{\lambda})$.
1: For $i = 1, \dots, K-1$, draw $u_i \sim U(0,1)$ and set $x_i = F^{-1}(x|\lambda_i, \lambda_K)$.
2: If $\sum_{i=1}^{K} x_i > 1$, return to step 1, otherwise return $\mathbf{u} = (u_1, \dots, u_{K-1})$.
```

C.2. The Ordered Rejection Sampler

An analysis of algorithm 1 reveals two relevant observations. Firstly, the simulation of each $x_j \sim \mathcal{CC}(\lambda_j, \lambda_K)$ in step 1 involves computing the inverse cdf $F^{-1}(\cdot|\lambda_i, \lambda_K)$, which is more expensive than computing the cumulative sum $\sum_{i=1}^{j} x_i$ of the draws. It therefore pays to recompute the cumulative sums after each draw and go directly to the rejection step as soon as it exceeds 1. Secondly, note that we do not generally expect the components of $\boldsymbol{\lambda}$ to be balanced. Thus, even though simulating each x_i in step 1 requires the same amount of computation, those drawn from smaller values of λ_i are more likely to be close to 0 than those drawn from higher values of λ_i. The dimensions that are more likely to be close to 0 are also less likely to make our cumulative sum exceed the rejection threshold. It therefore also pays to draw the x_i components in order of decreasing λ_i.

These remarks motivate an improved sampling scheme, which we call the ordered rejection sampler (algorithm 3). Empirically, we find that this sampler substantially reduces the rejection rate (see figure 1) as well as the computation time (by not only rejecting less, but also rejecting sooner). However, this sampler performs poorly when $\boldsymbol{\lambda}$ is balanced; such a setting leaves little room for improvement from the re-ordering operation, and the resulting sampler is similar to the naive rejection sampler. This motivates a further sampling scheme that we introduce in the following section, but further improvements to this sampler are left to future work. Lastly, we note that the reparameterization scheme of section C.1.1 can be modified trivially to apply here also.

Algorithm 3 Ordered rejection sampler

```
Input: target distribution $\mathcal{CC}(\boldsymbol{\lambda})$.
Output: sample $\mathbf{x}$ drawn from target.
1: Find the permutation $\pi$ that orders $\boldsymbol{\lambda}$ from largest to smallest, and let $\tilde{\boldsymbol{\lambda}} = \pi(\boldsymbol{\lambda})$.
2: Set the cumulative sum $c \leftarrow 0$ and $i \leftarrow 2$.
3: while $c < 1$ do
4:     Draw $u_i \sim U(0,1)$.
5:     Set $x_i = F^{-1}(u_i|\tilde{\lambda}_i, \tilde{\lambda}_1)$.
6:     Set $c \leftarrow c + x_i$.
7:     Set $i \leftarrow i + 1$.
8: end while
9: If $c > 1$, go back to step 2.
10: Set $x_1 = 1 - \sum_{i=2}^{K} x_i$.
11: Return $\mathbf{x} = \pi^{-1}(x_1, \dots, x_K)$.
```

C.3. The Permutation Sampler

Next, we develop a permutation sampler that performs particularly well for configurations of $\boldsymbol{\lambda}$ that are balanced (those that lead to distributions that are close to uniform). Our key insights here are that the unit cube can be partitioned into simplexes, each of which corresponds to a permutation of its dimensions, and that the CC distribution is, in a sense, 'invariant' over these permutations.

C.3.1. Partitioning the cube into simplexes

Let $\mathcal{R} = [0,1]^{K-1}$, the unit cube. For a permutation $\sigma : \{1, 2, \dots, K-1\} \to \{1, 2, \dots, K-1\}$, we denote $\mathcal{S}_\sigma = \{\mathbf{x} \in \mathbb{R}^{K-1} : 0 \le x_{\sigma(1)} \le x_{\sigma(2)} \le \cdots \le x_{\sigma(K-1)} \le 1\}$. We can then partition (up to intersections of Lebesgue measure zero) the cube using the $(K-1)!$ different permutations:

$$\mathcal{R} = \bigcup_\sigma \mathcal{S}_\sigma, \tag{18}$$

where the union is over all permutations. While our sample space cl($\mathbb{S}^{K-1}$), is not equal to $\mathcal{S}_\sigma$ for any σ, we will see in section C.3.2 that sampling from cl($\mathbb{S}^{K-1}$) and sampling from $\mathcal{S}_{id}$ are equivalent, where id is the identity permutation. However, as we will see in section C.3.3, sampling from $\mathcal{S}_{id}$ allows to take advantage of the cube partitioning of equation 18, while the same cannot be done for cl($\mathbb{S}^{K-1}$).

C.3.2. The equivalence of sampling over any simplex

In this section, we consider varying the support of our CC density from the standard simplex, to other simplexes as well as the unit cube. We denote the support explicitly by writing $\mathbf{x} \sim \mathcal{CC}_\mathcal{A}(\boldsymbol{\eta})$ for the density:

$$p_\mathcal{A}(\mathbf{x}|\boldsymbol{\eta}) \propto \exp\left(\sum_{i=1}^{K-1} \eta_i x_i\right) \mathbb{1}(\mathbf{x} \in \mathcal{A}) \tag{19}$$

where the subscript $\mathcal{A}$ will typically denote a simplex. Now, letting $\mathbf{x} \sim \mathcal{CC}_\mathcal{A}(\boldsymbol{\eta})$ and $\mathbf{y} = Q\mathbf{x}$, where $Q \in \mathbb{R}^{(K-1)\times(K-1)}$ is an invertible matrix, it follows by the change of variable formula that:

$$\begin{aligned} p_{Q(\mathcal{A})}(\mathbf{y}|\boldsymbol{\eta}) &= \frac{1}{|\det(Q)|} p_\mathcal{A}(Q^{-1}\mathbf{y}|\boldsymbol{\eta}) \\ &\propto \exp(\boldsymbol{\eta}^\top [Q^{-1}\mathbf{y}]) \mathbb{1}(y \in Q(\mathcal{A})) \\ &= \exp([Q^{-\top}\boldsymbol{\eta}]^\top \mathbf{y}) \mathbb{1}(y \in Q(\mathcal{A})), \end{aligned} \tag{20}$$

where $Q(\mathcal{A}) = \{\mathbf{y} : \mathbf{y} = Q\mathbf{x}, \mathbf{x} \in \mathcal{A}\}$. Thus, we have that $\mathbf{y} \sim \mathcal{CC}_{Q(\mathcal{A})}(\tilde{\boldsymbol{\eta}})$, where $\tilde{\boldsymbol{\eta}} = Q^{-\top}\boldsymbol{\eta}$, so that $\mathbf{y}$ has a new CC distribution on a transformed sample space. Moreover, if Q is a permutation matrix and $\mathcal{A} = \mathcal{S}_\sigma$ for some permutation σ, then $Q(\mathcal{A})$ is a 'permuted' simplex, and $\tilde{\boldsymbol{\eta}}$ is a rearranged parameter vector, hence the equivalence of sampling over any simplex for the CC.

C.3.3. The permutation sampling algorithm

Now, consider a lower triangular matrix of ones:

$$B = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 0 & \cdots & 0 \\ 1 & 1 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 1 & \cdots & 1 \end{pmatrix}. \tag{21}$$

Note that $\mathcal{S}_{id} = B(\text{cl}(\mathbb{S}^{K-1}))$, so that sampling from $\mathcal{CC}_{\text{cl}(\mathbb{S}^{K-1})}(\boldsymbol{\eta})$ is equivalent to sampling from $\mathcal{CC}_{\mathcal{S}_{id}}(\tilde{\boldsymbol{\eta}})$ and transforming the result with B^{-1}, where $\tilde{\boldsymbol{\eta}} = B^{-\top}\boldsymbol{\eta}$. Now consider rejection sampling to draw from $\mathcal{CC}_{\mathcal{S}_{id}}(\tilde{\boldsymbol{\eta}})$. As with the naive sampler, our proposal can be drawn on the whole unit cube from independent 1-dimensional CB variates, but the advantage here is that we do not have to directly reject the sample if it fell outside of the desired simplex $\mathcal{S}_{id}$, but rather we can transform it onto that simplex and then accept it with an appropriate probability (which we can compute easily using the invariance property). Here, the acceptance probability depends on which simplex the proposal fell into, and is given by:

$$\alpha(\mathbf{y}, \tilde{\boldsymbol{\eta}}, P) = \frac{p_{\mathcal{S}_{id}}(\mathbf{y}|\tilde{\boldsymbol{\eta}})}{\kappa(\tilde{\boldsymbol{\eta}}, P) p_{\mathcal{S}_{id}}(\mathbf{y}|P^{-\top}\tilde{\boldsymbol{\eta}})}, \tag{22}$$

where $\kappa(\tilde{\boldsymbol{\eta}}, P)$ is the rejection sampling constant, which in this case is equal to:

$$\kappa(\tilde{\boldsymbol{\eta}}, P) = \max_{\mathbf{y}\in\mathcal{S}_{id}} \frac{p_{\mathcal{S}_{id}}(\mathbf{y}|\tilde{\boldsymbol{\eta}})}{p_{\mathcal{S}_{id}}(\mathbf{y}|P^{-\top}\tilde{\boldsymbol{\eta}})}. \tag{23}$$

Algorithm 4 Permutation sampler

```
Input: target distribution $\mathcal{CC}(\boldsymbol{\eta})$.
Output: sample $\mathbf{x}$ drawn from target.
1: Sample $\mathbf{y}' \sim \mathcal{CC}_{\mathcal{R}}(\tilde{\boldsymbol{\eta}})$ (again, this is straightforward to do by sampling each coordinate independently).
2: Sort the elements of $\mathbf{y}'$. In other words, find a permutation $\sigma$ such that $\sigma(\mathbf{y}') \in \mathcal{S}_{id}$. Let $P$ be the corresponding permutation matrix and $\mathbf{y} = P\mathbf{y}'$.
3: Compute $\kappa(\tilde{\boldsymbol{\eta}}, P)$ by taking the maximum over the the vertices of $\mathcal{S}_{id}$, and use this to compute the acceptance probability $\alpha(\mathbf{y}, \tilde{\boldsymbol{\eta}}, P)$.
4: Accept $\mathbf{y}$ with probability $\alpha(\mathbf{y}, \tilde{\boldsymbol{\eta}}, P)$. Otherwise, go back to step 1.
5: Return $\mathbf{x} = B^{-1}\mathbf{y}$.
```

The algorithm samples correctly from $\mathcal{CC}_{\mathcal{S}_{id}}(\tilde{\boldsymbol{\eta}})$, because $\mathbf{y}'$ can be thought of as a sample from $p_{\mathcal{R}}(\mathbf{y}'|\boldsymbol{\eta}, \mathbf{y}' \in \mathcal{S}_\sigma) = p_{\mathcal{S}_\sigma}(\mathbf{y}'|\boldsymbol{\eta})$, which we then transform with P to obtain a distribution on $\mathcal{S}_{id}$. If we use this distribution as a proposal distribution for a rejection sampling algorithm, we recover precisely the acceptance probability of equation 22. Intuitively, if our sample $\mathbf{x}$ does not fall on the desired simplex $\mathcal{S}_{id}$, we move around the simplex in which it fell (along with $\mathbf{x}$ itself) so that it matches the desired simplex, and then do rejection sampling.

We conclude this subsection with two short notes on the optimization problem of equation 23. The first one is that when $\boldsymbol{\eta}^{(2)} = Q^{-\top}\boldsymbol{\eta}^{(1)}$ where $|\det(Q)| = 1$, then the normalizing constants cancel out, which is the case in our algorithm since $|\det(P)| = 1$. The second is that, by taking logs, the optimization problem can be transformed into a linear problem subject to linear inequality constraints, meaning that the solution must be achieved at a vertex. Since there are K vertices, namely $\mathbf{0}$, $\mathbf{e}_{K-1}, \mathbf{e}_{K-1} + \mathbf{e}_{K-2}, \ldots, \sum_{i=1}^{K-1} \mathbf{e}_i$, we can solve the problem by simply checking each of these vertices.

C.4. Performance

While the ordered rejection sampler can never have a worse rejection rate than its naive counterpart, the comparison with the permutation sampler depends on the shape of the target distribution, as discussed. The perfectly balanced case $\boldsymbol{\lambda} = (1/K, \ldots, 1/K)$ results in the worst possible rejection rate for the ordered rejection sampler (we accept with probability $1/(K-1)!$), but also the best possible rejection rate for the permutation sampler (this is the uniform case so $\alpha(\mathbf{y}, \tilde{\boldsymbol{\eta}}, P) = 1$). On the other end of the spectrum, in the totally unbalanced case where one element of $\boldsymbol{\lambda}$ holds all the weight and the others are close to zero, the ordered rejection sampler achieves an acceptance rate close to 1, whereas it is much smaller for the permutation sampler (see section C.4.3). In this sense, our samplers are complementary, and an optimal sampling algorithm could involve combining accept/reject steps from both methods. We study the performance of our samplers empirically, by comparing the distribution of the rejection rates under a sparsity-inducing prior $\boldsymbol{\lambda} \sim Dirichlet(1/K, \ldots, 1/K)$. Indeed, the ordered rejection sampler tends to considerably outperform the permutation sampler (see figure 1), as well as (trivially) the naive sampler.

Now, it may come as a surprise that the permutation sampler does not necessarily outperform the naive sampler. After all, the naive sampler only accepts samples that fell directly into the desired simplex, whereas the permutation sampler has the additional possibility of accepting a sample that fell outside of $\mathcal{S}_{id}$ after applying a suitable permutation. However, this intuition breaks down once we realize that the proposal distributions from the two methods are not equivalent. We make this precise in the following sections.

C.4.1. Rejection rate - naive sampler

Suppose we seek $\mathbf{x} \sim \mathcal{CC}_{\text{cl}(\mathbb{S}^{K-1})}(\boldsymbol{\eta})$. The naive rejection sampler proposes $\mathbf{x} \sim \mathcal{CC}_{\mathcal{R}}(\boldsymbol{\eta})$, and accepts if $\mathbf{x} \in \mathbb{S}^{K-1}$. The proposal density is equal to (we know the normalizing constant as we have the product of independent CBs):

$$p_{\mathcal{R}}(\mathbf{x}|\boldsymbol{\eta}) = \prod_{i=1}^{K-1} \frac{\eta_i}{e^{\eta_i} - 1} e^{\eta_i x_i}. \tag{24}$$

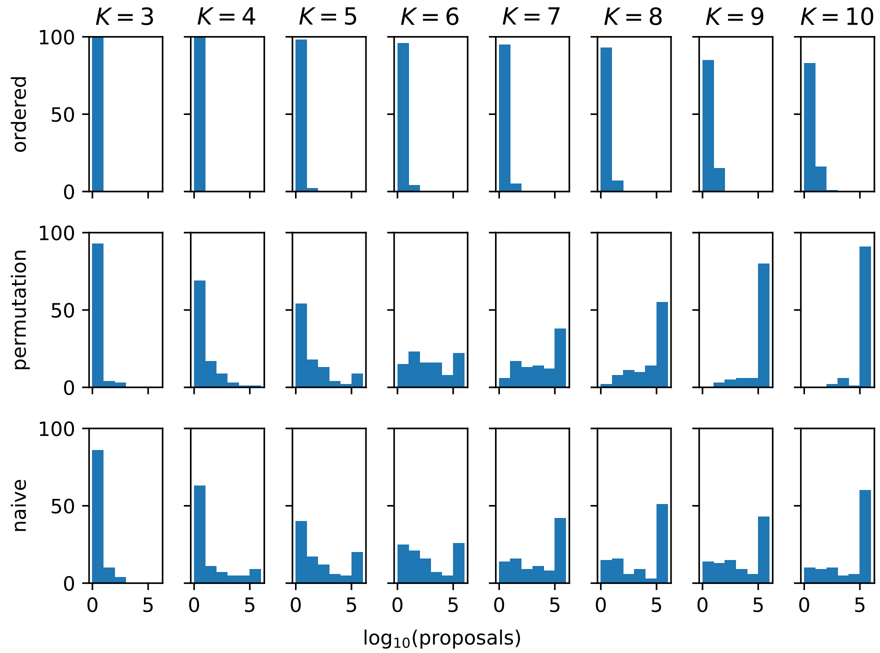

Figure 1. Shows the performance of 3 sampling algorithms across different dimensions K. Each histogram shows the distribution, over 100 trials, of the number of proposals required for 1 acceptance, on the log scale (base 10). The distributions are not exponential, since each of the 100 trials is sampled from a different $\mathcal{CC}(\boldsymbol{\lambda})$ distribution, where the parameter follows $\boldsymbol{\lambda} \overset{iid}{\sim} Dirichlet(1/K, \cdots, 1/K)$. Due to computational constraints, the number of proposals in each trial is right-censored, hence the large bars at the right end of the histograms.

Therefore, the probability of acceptance is:

$$P(\mathcal{CC}_{\mathcal{R}}(\boldsymbol{\eta}) \in \mathbb{S}^{K-1}) = \int_{\mathbb{S}^{K-1}} \prod_{i=1}^{K-1} \frac{\eta_i}{e^{\eta_i} - 1} e^{\eta_i x_i} d\mu. \tag{25}$$

We can apply the transformation B to rewrite this as:

$$P(B(\mathcal{CC}_{\mathcal{R}}(\boldsymbol{\eta})) \in B(\mathbb{S}^{K-1})) = P(\mathcal{CC}_{B(\mathcal{R})}(B^{-\top}\boldsymbol{\eta}) \in S_{id}) = \int_{\mathcal{S}_{id}} \prod_{i=1}^{K-1} \frac{\eta_i}{e^{\eta_i} - 1} e^{\tilde{\eta}_i x_i} d\mu, \tag{26}$$

where we have used the fact that $|\det(B)| = 1$ so that the normalizing constant remains unchanged. Thus, the probability of acceptance of the naive sampler is equal to:

$$P(accept) = \left(\prod_{i=1}^{K-1} \frac{\eta_i}{e^{\eta_i} - 1} \right) \cdot \int_{\mathcal{S}_{id}} e^{\tilde{\boldsymbol{\eta}}^\top \mathbf{x}} d\mu. \tag{27}$$

C.4.2. Rejection rate - permutation sampler

In the case of the permutation sampler, the acceptance rate is harder to compute. However, one can easily obtain a lower bound by considering only the samples that fall directly into our target simplex (in this case, $\mathcal{S}_{id}$). In this case, the proposal distribution is:

$$p_{\mathcal{R}}(\mathbf{x}|\tilde{\boldsymbol{\eta}}) = \prod_{i=1}^{K-1} \frac{\tilde{\eta}_i}{e^{\tilde{\eta}_i} - 1} e^{\tilde{\eta}_i x_i}. \tag{28}$$

If the resulting sample falls in $\mathcal{S}_{id}$, we accept the sample and map it back to cl($\mathbb{S}^{K-1}$). Thus, the acceptance rate has the lower bound

$$P(accept) \geq \left(\prod_{i=1}^{K-1} \frac{\tilde{\eta}_i}{e^{\tilde{\eta}_i} - 1}\right) \cdot \int_{\mathcal{S}_{id}} e^{\tilde{\boldsymbol{\eta}}^\top \mathbf{x}} d\mu. \tag{29}$$

Note that, while this lower bound has the same integral term as the naive rejection sampler, it is multiplied by a different normalizing constant. In particular, there are configurations of $\boldsymbol{\eta}$ that can lead to much worse normalizing constants for the permutation sampler than for the naive rejection sampler, resulting in a worse acceptance rate overall.

C.4.3. EXAMPLE

We give an example of a configuration of $\boldsymbol{\eta}$ such that the acceptance rate of the naive sampler is better than that of the permutation sampler. Consider the case $(\eta_1, \cdots, \eta_{K-1}) = (-M, \cdots, -M)$, where M is a large positive number. Note that this example is far from the uniform case $\boldsymbol{\lambda} = (1/K, \cdots, 1/K)$. In fact, in this case, after transforming with B, we obtain $\tilde{\eta}_{K-1} = -M$, and $\tilde{\eta}_{K-2} = \cdots = \tilde{\eta}_1 = 0$. Thus, when we sample a proposal $\mathbf{y} \sim \mathcal{CC}_{\mathcal{R}}(\tilde{\boldsymbol{\eta}})$, typically y_{K-1} will be small relative to $y_1, \cdots, y_{K-2}$, which will be ordered at random. This means the sorting step of our permutation sampler will likely map y_{K-1} to y'_1, as well as sorting the remaining entries $y'_2 < \cdots < y'_{K-1}$. In other words, P maps the $(K-1)^{th}$ entry to the 1st entry, and one of the first $K-2$ entries to the $(K-1)^{th}$ entry (whichever of these happens to sample the largest value). The resulting distributions $p_{\mathcal{S}_{id}}(\cdot|\tilde{\boldsymbol{\eta}})$ and $p_{\mathcal{S}_{id}}(\cdot|P^{-\top}\tilde{\boldsymbol{\eta}})$ are similar, with the key difference that the former puts the negative $\tilde{\boldsymbol{\eta}}$ coefficient (namely $\tilde{\eta}_{K-1} = -M$) in the last position (the largest component of $\mathbf{y}$), while the latter puts it into some other position determined by σ^{-1}, i.e. the right-most column of P or equivalently, the bottom row of P^{-1}. In this setting, it follows that κ will be equal to 1, and the ratio $p_{\mathcal{S}_{id}}(\mathbf{y}'|\tilde{\boldsymbol{\eta}})/p_{\mathcal{S}_{id}}(\mathbf{y}'|P^{-\top}\tilde{\boldsymbol{\eta}})$ will be small. Thus, our rejection sampling ratio is typically small and we are likely to reject our proposal. The ratio will be close to 1 only in the event that the proposed value x_{K-1} is large relative to the other components $x_1, \cdots, x_{K-2}$, which rarely happens as x_{K-1} is sampled from a univariate CC with much smaller coefficient. Note further, that $\tilde{\boldsymbol{\eta}}$ cannot be re-shuffled in this case, as this would lead to a target distribution on a simplex other than $\mathcal{S}_{id}$ (we can only shuffle $\boldsymbol{\eta}$ prior to applying B, which in this case leaves $\boldsymbol{\eta}$ unchanged). We conclude that we cannot achieve a uniformly better rejection rate through the permutation sampler, relative to the naive method.

References

Connor, R. J. and Mosimann, J. E. Concepts of independence for proportions with a generalization of the dirichlet distribution. *Journal of the American Statistical Association*, 64(325):194–206, 1969.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

Loaiza-Ganem, G. and Cunningham, J. P. The continuous bernoulli: fixing a pervasive error in variational autoencoders. In *Advances in Neural Information Processing Systems*, pp. 13266–13276, 2019.

Naesseth, C. A., Ruiz, F. J. R., Linderman, S. W., and Blei, D. M. Reparameterization gradients through acceptance-rejection sampling algorithms. In *International Conference on Artificial Intelligence and Statistics*, pp. 489–498, 2017.

Paisley, J. W., Zaas, A. K., Woods, C. W., Ginsburg, G. S., and Carin, L. A stick-breaking construction of the beta process. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pp. 847–854, 2010.