Improved control of Dirichlet location and scale near the boundary

Catherine Xue[*] Alessandro Zito[†], and Jeffrey W. Miller[‡]

Abstract. Dirichlet distributions are commonly used for modeling vectors in a probability simplex. When used as a prior or a proposal distribution, it is natural to set the mean of a Dirichlet to be equal to the location where one wants the distribution to be centered. However, if the mean is near the boundary of the probability simplex, then a Dirichlet distribution becomes highly concentrated either (i) at the mean or (ii) extremely close to the boundary. Consequently, centering at the mean provides poor control over the location and scale near the boundary. In this article, we introduce a method for improved control over the location and scale of Beta and Dirichlet distributions. Specifically, given a target location point and a desired scale, we maximize the density at the target location point while constraining a specified measure of scale. We consider various choices of scale constraint, such as fixing the concentration parameter, the mean cosine error, or the variance in the Beta case. In several examples, we show that this maximum density method provides superior performance for constructing priors, defining Metropolis-Hastings proposals, and generating simulated probability vectors.

Keywords: Beta distribution, Bayesian inference, Conjugate priors, Dirichlet distribution, Markov chain Monte Carlo.

[*]Department of Biostatistics, Harvard University, csxue@g.harvard.edu
[†]Department of Biostatistics, Harvard University, azito@hsph.harvard.edu
[‡]Department of Biostatistics, Harvard University, jwmiller@hsph.harvard.edu

1 Introduction

The family of Dirichlet distributions is the default choice when modeling probability vectors, due to its flexibility and analytical tractability. As a conjugate prior for the parameters of a multinomial distribution, the Dirichlet plays a central role in many Bayesian models such as latent Dirichlet allocation (Blei et al., 2003), finite and infinite mixture models (Rousseau and Mengersen, 2011; Richardson and Green, 1997; Miller and Harrison, 2018), admixture models (Pritchard et al., 2000), non-negative matrix factorization (Zito and Miller, 2024), and variable selection with global-local shrinkage (Bhattacharya et al., 2015; Zhang and Bondell, 2018).

When specifying the parameters of a Dirichlet$(\alpha u_1, \ldots, \alpha u_K)$ distribution, where $\sum_{i=1}^{K} u_i = 1$ and $\alpha > 0$, the standard approach is to set the mean $(u_1, \ldots, u_K)$ equal to the location where one would like the distribution to be centered and adjust the concentration parameter α to control the scale. Unfortunately, if the mean is close to the boundary of the probability simplex—that is, if one or more entries is near zero—then this approach exhibits serious problems. First, if the mean is near the boundary, then α provides little control over the scale of the distribution. Naturally, for large values of α, the distribution is highly concentrated around the mean. Intuitively, one might expect that using a smaller concentration would lead to a less informative prior. However, as α decreases, the distribution does not become much more spread out – and, in fact, the bulk of the mass just moves even farther towards the boundary, becoming concentrated extremely close to it. As a result, the mean ends up being so far in the tail that it is no longer a useful representation of the center of the distribution. Figure 1 illustrates this behavior in the case of a Beta distribution, which is representative of the general Dirichlet case since the marginals of a Dirichlet are Beta-distributed. Thus, the mean and concentration parameter are not useful for controlling the location and scale of Dirichlet distributions near the boundary.

This pathological behavior may have severe consequences. Some Bayesian models using Beta or Dirichlet priors may unintentionally be forcing probabilities to be essentially zero, even when the prior mean is not that close to zero. Metropolis–Hastings proposals using Beta or Dirichlet distributions may lead to very poor mixing because the proposals are extremely close to the boundary with high probability, rather than being near the current state. Furthermore, the problem is exacerbated in high dimensions: In a high-dimensional probability simplex, every point is near the boundary because the sum-to-one constraint forces many entries to be close to zero.

In this paper, we propose a novel method for specifying the parameters of a Dirichlet distribution in a way that provides better control over the location and scale. Specifically, given a target location $c = (c_1, \ldots, c_K)$ and a scale parameter s, we maximize the density at c subject to the constraint that a specified measure of scale is equal to s. For instance, the choice of scale may be the concentration parameter, the sum of the variances, the mean cosine error, or some other value quantifying the spread of the distribution. This *maximum density* approach has several attractive features. First, it provides greater control over the scale of the distribution. Additionally, it tends to put more probability mass near the target location c, because it maximizes the density at c by construction.
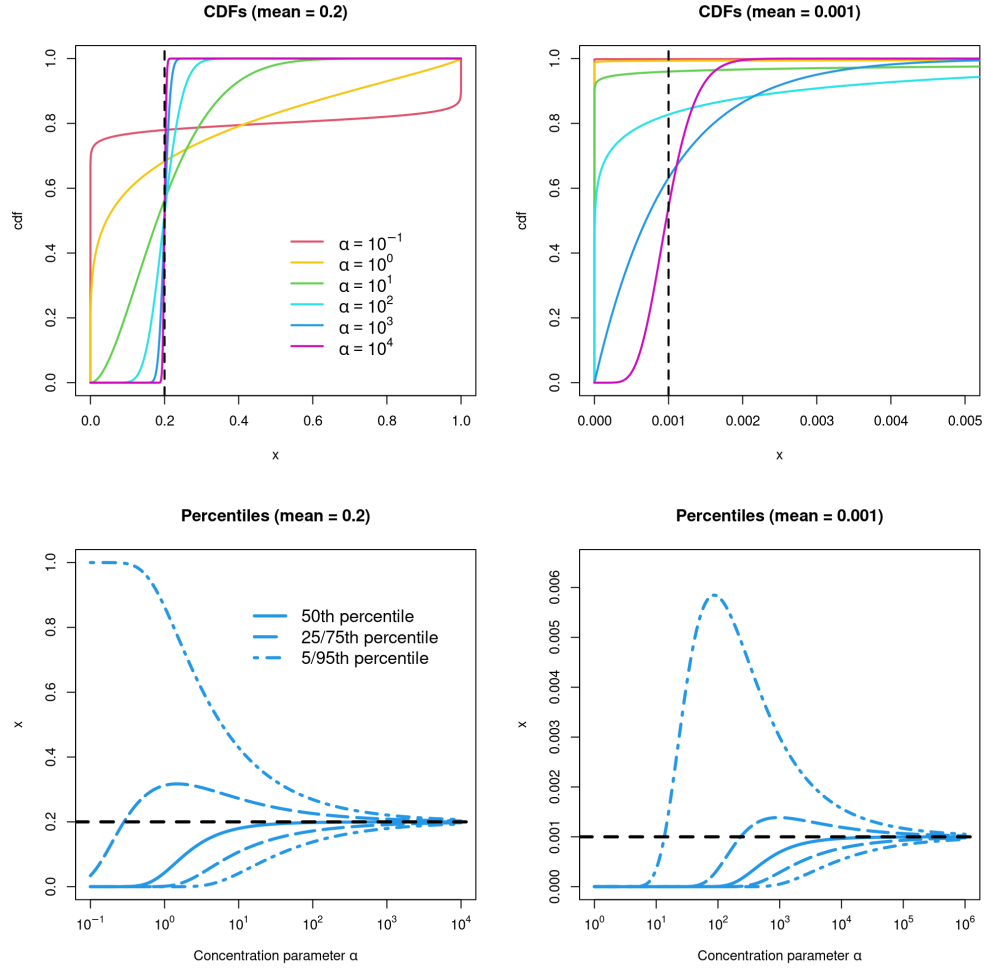
Figure 1: Setting the mean equal to the desired location is problematic near the boundary. (*Top*) Cumulative distribution functions (CDFs) of Beta distributions with means 0.2 and 0.001, for a range of concentration parameters α. The mean of $\mathrm{Beta}(a, b)$ is $a/(a + b)$ and the concentration parameter is $\alpha = a + b$. (*Bottom*) Percentiles of the same distributions, as a function of α. Even when the mean is not close to the boundary (such as 0.2), most of the mass just shifts from the mean to the boundary as α goes from high to low. When the mean is close to the boundary (such as 0.001), the distribution is quite concentrated for all α, and α provides very little control over the scale.

Furthermore, it can be computed using a fast and simple algorithm for performing the constrained optimization.

The rest of the article is organized as follows. In Section 2, we describe our proposed methodology. In Section 3, we illustrate the method's utility for (i) Metropolis–Hastings proposals with improved mixing near the boundary, (ii) Bayesian inference for the probability of rare events, and (iii) generating random probability vectors for mutational signatures analysis in cancer genomics. We conclude with a brief discussion in Section 4.

2 Methodology

In this section, we introduce the maximum density method. We first consider the special case of Beta distributions (Section 2.1), then generalize to Dirichlet distributions (Section 2.2), and provide a step-by-step algorithm (Section 2.3).

2.1 Maximum density method for specifying Beta distributions

The Beta distribution with parameters $a > 0$ and $b > 0$, denoted $\text{Beta}(a, b)$, has density $\text{Beta}(x \mid a, b) = x^{a-1}(1-x)^{b-1}/\text{B}(a, b)$ for $x \in (0, 1)$, where $\text{B}(a, b)$ is the beta function. Given a target location $c \in (0, 1)$, we propose to set a and b via the optimization:

$$\arg\max_{a,b>0} \text{Beta}(c \mid a, b) \text{ subject to } h(a, b) = 0, \tag{1}$$

where $h(a, b) = 0$ represents a desired scale constraint. In the Beta case, we focus particularly on constraining the variance to a given value v by defining $h(a, b) = V(a, b)/v - 1$, where $V(a, b) = ab/\big((a + b)^2(a + b + 1)\big)$ is the variance of $\text{Beta}(a, b)$. In other words, out of all the Beta distributions with variance v, we choose the one with highest density at the target location c. We refer to this as the *maximum density* method.

For $v \in (0, 1/4)$, the solution to this optimization always exists and is finite, as we show in Theorem 1. To compute the solution, we develop an algorithm based on Newton's method with equality constraints; see Section 2.3. This algorithm reliably converges to the constrained maximum for any $c \in (0, 1)$, $v \in (0, 1/4)$. The convergence is rapid for $v < 0.2$, and becomes slower as v approaches $1/4$.

Theorem 1. *For all $c \in (0, 1)$ and $v \in (0, 1/4)$, there exists a finite solution to:*

$$\arg\max_{a,b>0} \text{Beta}(c \mid a, b) \text{ subject to } V(a, b) = v,$$

where $V(a, b) = ab/\big((a + b)^2(a + b + 1)\big)$.

All proofs are provided in Section S1 of the Supplementary Material. In Theorem 1, the restriction to $v \in (0, 1/4)$ is necessary since there do not exist Beta distributions with variance greater or equal to $1/4$, as we show in Theorem 2.

Theorem 2. *There is a Beta distribution with mean u and variance v if and only if $0 < v < 1/4$ and $|u - 1/2| < (1/2)\sqrt{1 - 4v}$.*

Fixing the concentration parameter instead of the variance

In some situations, it may be preferable to constrain the concentration $\alpha = a + b$ instead of the variance. In this case, the maximum density method is to set a and b via:

$$\underset{a,b>0}{\arg\max} \operatorname{Beta}(c \mid a, b) \text{ subject to } a + b = \alpha, \qquad (2)$$

given $c \in (0, 1)$ and $\alpha > 0$. Theorem 1 extends to this case as well, that is, for all $c \in (0, 1)$ and $\alpha \in (0, \infty)$ there is a finite solution to Equation 2; see Theorem S1. A slightly different version of the Newton's method algorithm can be used to solve this; see Section 2.3. In Figures 2 and 3, we use this fixed α version (rather than fixed variance) to facilitate comparisons with the mean method with fixed α.

Comparison of the distributions attained by each method

The maximum density method resolves the issues with the mean method seen in Figure 1. As shown in Figure 2, the maximum density method exhibits a greater range of control over the scale of the distribution, while keeping the distribution more centered at the target location in terms of percentiles.

To quantify how far the mass of the distribution is from the target location c, we consider $Y = |\operatorname{logit}(X) - \operatorname{logit}(c)|$, where $\operatorname{logit}(x) = \log(x/(1-x))$ and $X \sim \operatorname{Beta}(a, b)$. Small values of Y mean that X is close to c on the logit scale. Transforming to the logit scale makes it possible to evaluate differences in magnitude close to the boundary. Figure 3 shows the density of Y when using (i) the mean method and (ii) our maximum density method; for both methods we compare results when using $c \in \{0.001, 0.2\}$ and $\alpha \in \{0.1, 1, 10\}$. More precisely, the mean method chooses $a = \alpha c$ and $b = \alpha(1 - c)$. For the maximum density method, we use target location c and concentration α. We derive a closed-form expression for the density of Y in Section S2.

Figure 3 demonstrates that the maximum density method puts more probability mass near the target location c, compared to the mean method. Specifically, we see that the distribution of Y has more probability mass near 0 under the maximum density method, meaning that X is closer to c with high probability. The difference is especially stark near the boundary, for instance when $c = 0.001$.

Median method: An alternative approach

An alternative to our maximum density approach would be to choose a Beta distribution with median equal to the target location c and variance equal to v (or concentration parameter equal to α). In the case of Beta distributions, this works reasonably well. However, the maximum density approach has three advantages: (1) it tends to put more mass near the target location c, (2) optimization is more tractable and stable, and (3) it extends more naturally to the general Dirichlet case. See Section S4 for more details on this median-based approach.
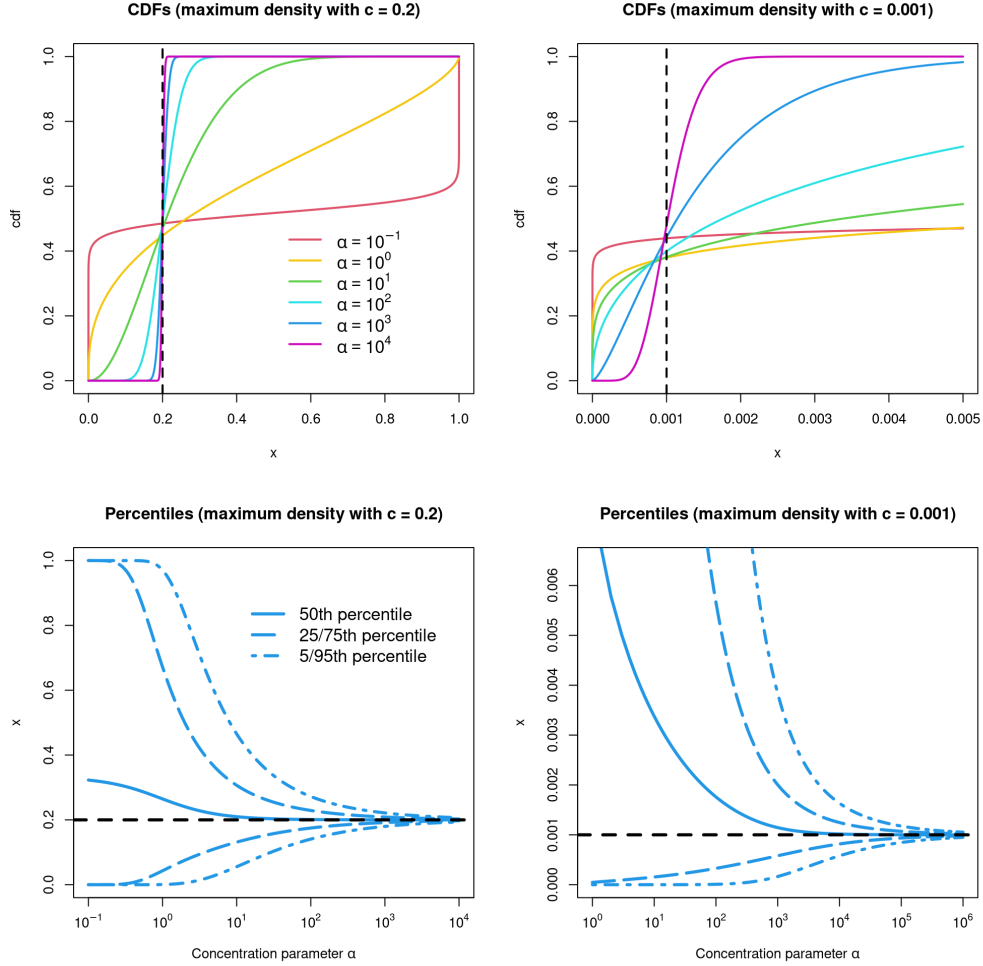
Figure 2: Maximum density provides better control of location and scale near the boundary. (*Top*) CDFs of Beta distributions determined by the maximum density method with target locations $c = 0.2$ and $c = 0.001$, for the same range of concentration parameters α as in Figure 1; here, we maximize $\text{Beta}(c \mid a, b)$ subject to $a + b = \alpha$ to enable comparison with Figure 1, see Section 2.3 for details. (*Bottom*) Percentiles of the same distributions, as a function of α. A wide range of scales can be attained by varying α (or correspondingly, varying v) when using the maximum density method, regardless of whether the target location c is near the boundary. Furthermore, c remains between the first quartile and the median of the distribution as α decreases, rather than moving into the upper tail as in Figure 1.
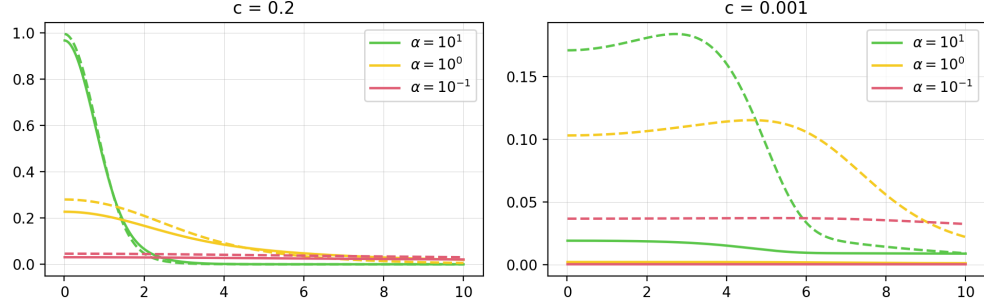
Figure 3: For Beta distributions, the maximum density method puts more probability mass near the target location, compared to the mean method. The plots show the density of $Y = |\mathrm{logit}(X) - \mathrm{logit}(c)|$ when X is Beta distributed with parameters chosen by either (i) setting the mean equal to c (solid line), or (ii) using the maximum density method with target location c (dashed line). For both methods, we use concentration parameters of $\alpha = 10$, $\alpha = 1$, and $\alpha = 0.1$.

2.2   Maximum density method for specifying Dirichlet distributions

The Dirichlet distribution with parameters $a_1, \ldots, a_K > 0$ has density

$$\mathrm{Dirichlet}(x \mid a_1, \ldots, a_K) = \frac{\Gamma\big(\sum_{i=1}^{K} a_i\big)}{\prod_{i=1}^{K} \Gamma(a_i)} \, x_1^{a_1 - 1} \cdots x_K^{a_K - 1} \tag{3}$$

for $x = (x_1, \ldots, x_K) \in \Delta_K$, where $\Delta_K := \big\{ x \in \mathbb{R}^K : x_1, \ldots, x_K > 0, \sum_{i=1}^{K} x_i = 1 \big\}$ is the probability simplex. Here, $\Gamma(x)$ denotes the gamma function.

Letting $X \sim \mathrm{Dirichlet}(a_1, \ldots, a_K)$, it can be shown that the ith coordinate X_i follows a Beta distribution, specifically, $X_i \sim \mathrm{Beta}(a_i, \sum_{j \neq i} a_j)$. Thus, the Beta distribution can essentially be thought of as the special case of a Dirichlet with $K = 2$, although technically the Beta corresponds to one coordinate of a Dirichlet. The mean of the ith coordinate is $\mathbb{E}(X_i) = a_i / \sum_{j=1}^{K} a_j$ and the scale of the Dirichlet distribution around its mean is traditionally thought to be controlled by the *concentration parameter*, $\alpha = \sum_{i=1}^{K} a_i$. However, when the mean of X_i is near zero or one, the distribution of X_i exhibits the same pathologies as in the case of a Beta distribution. This follows simply because X_i is, in fact, Beta distributed. In particular, the concentration parameter α exhibits little control over the scale, and the distribution of X_i concentrates near the boundary as α decreases.

We extend our maximum density method to the general case of a K-dimensional Dirichlet distribution as follows. Suppose $c = (c_1, \ldots, c_K) \in \Delta_K$ is a target location in the probability simplex. We propose to choose the Dirichlet parameters $a_1, \ldots, a_K$ via:

$$\underset{a_1, \ldots, a_K > 0}{\arg\max} \ \mathrm{Dirichlet}(c \mid a_1, \ldots, a_K) \ \text{subject to} \ h(a_1, \ldots, a_K) = 0, \tag{4}$$
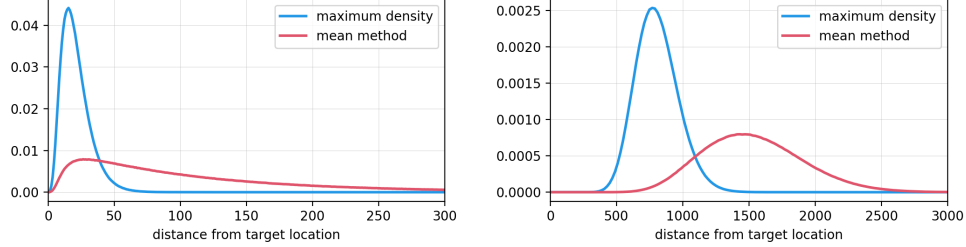
Figure 4: For Dirichlet distributions, the maximum density method puts more probability mass closer the target location. The plots show the density of $Y = \sum_{i=1}^{K} |\text{logit}(X_i) - \text{logit}(c_i)|$ when $X \sim \text{Dirichlet}(a_1, \ldots, a_K)$ with $a_1, \ldots, a_K$ chosen by either (i) setting the mean equal to c (red line), or (ii) using the maximum density method with target location c (blue line). The concentration parameter is $\alpha = 1$ for both methods. (*Left*) $c = (0.01, 0.1, 0.2, 0.3, 0.39)$; (*Right*) $c = (1, 2, 3, \ldots, 30)/\sum_{i=1}^{30} i$.

where h represents a desired constraint. This is a direct generalization of Equation 1. We provide an algorithm for solving this optimization problem in Algorithm 1. One simple choice of constraint is to fix the concentration parameter to a given value α; we do this by defining $h(a_1, \ldots, a_K) = (\sum_{i=1}^{K} a_i)/\alpha - 1$. Another option would be to control the variances, however, in the multivariate case we have to summarize the scale of the distribution with a single number. Motivated by an application to mutational signatures, we consider controlling the mean cosine error between X and $\mathbb{E}(X)$; see Sections 3.3 and S3 for details.

Figure 4 shows that for Dirichlet distributions, the maximum density method puts more mass near c compared to setting the mean equal to c; compare with Figure 3 in the Beta case. Here, we constrain α to enable direct comparison with the mean method.

2.3   Optimization algorithm for the maximum density method

To solve the constrained optimization problem for the maximum density method (Equations 1 and 4), we use an algorithm based on Newton's method with equality constraints (Boyd and Vandenberghe, 2004). This generalization of Newton's method is a second-order optimization technique for problems of the form

$$\arg\min_{x} f(x) \text{ subject to } h(x) = 0,$$

where f is a strictly convex, twice-differentiable function and h is differentiable. We provide versions of this algorithm for implementing the maximum density method with various choices of constraint function; see Section S3 for the derivation.

Algorithm 1 provides a step-by-step procedure for the general case of Dirichlet distributions; see Algorithm S1 for the special case of Beta distributions. As default settings,

---

Algorithm 1 Maximum density method for Dirichlet distributions

---

Input: target location $c \in \Delta_K$, step size $\rho \in (0,1]$, maximum number of iterations `maxiter` > 0, and convergence tolerance `tol` > 0.

1: $a_i \leftarrow 10(c_i + 1)/2$ for $i \in \{1, \ldots, K\}$ $\qquad\qquad\qquad\qquad$ ▷ Initialization
2: **for** iter $= 1, \ldots,$ `maxiter` **do**
3: $\qquad s \leftarrow a_1 + \cdots + a_K$
4: $\qquad g_i \leftarrow \psi(a_i) - \psi(s) - \log(c_i)$ for $i \in \{1, \ldots, K\}$ $\qquad$ ▷ Compute gradient
5: $\qquad H_{ij} \leftarrow \psi'(a_i)\mathbb{1}(i = j) - \psi'(s)$ for $i, j \in \{1, \ldots, K\}$ $\quad$ ▷ Compute Hessian matrix
6: $\qquad h \leftarrow h(a)$ $\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ ▷ Value of constraint
7: $\qquad J_i \leftarrow \partial h / \partial a_i$ for $i \in \{1, \ldots, K\}$ $\qquad\qquad\qquad$ ▷ Jacobian of constraint
8: $\qquad$ Solve for $\delta \in \mathbb{R}^K$ and $\lambda \in \mathbb{R}$ in the linear system: $\qquad$ ▷ Compute Newton step

$$\begin{bmatrix} H & J^{\mathrm{T}} \\ J & 0 \end{bmatrix} \begin{bmatrix} \delta \\ \lambda \end{bmatrix} = \begin{bmatrix} -g \\ -h \end{bmatrix}$$

$\qquad$ where $H = [H_{ij}] \in \mathbb{R}^{K \times K}$, $J = [J_1, \ldots, J_K] \in \mathbb{R}^{1 \times K}$, and $g = [g_i] \in \mathbb{R}^{K \times 1}$.
9: $\qquad a' \leftarrow a$ $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ ▷ Store current values
10: $\qquad a \leftarrow a' + \rho\,\delta$ $\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ ▷ Update values
11: $\qquad$ for $i \in \{1, \ldots, K\}$, if $a_i \leq 0$ then $a_i \leftarrow a_i'/2$ $\qquad$ ▷ Enforce boundary constraints
12: $\qquad$ **if** $|h| + \sum_{i=1}^{K} |a_i/a_i' - 1| <$ `tol` **then** $\qquad\qquad$ ▷ Check for convergence
13: $\qquad\qquad$ **output** a $\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ ▷ Return output
14: $\qquad$ **end if**
15: **end for**

Output: Dirichlet parameters $a = (a_1, \ldots, a_K)$.

---

we use $\rho = 1/2$, `maxiter` $= 100$, and `tol` $= 10^{-8}$. If Algorithm 1 reaches `maxiter` iterations without converging, then we set $\rho \leftarrow \rho/5$ and `maxiter` $\leftarrow 5\,$`maxiter`, and run the algorithm again; if it still fails to converge after 5 such restarts, we stop. In Algorithm 1, $\psi(x)$ and $\psi'(x)$ are the digamma and trigamma functions, that is, the first and second derivatives of $\log\Gamma(x)$. Also, $\mathbb{1}(\cdot)$ is the indicator function, that is, $\mathbb{1}(E) = 1$ if E is true, and $\mathbb{1}(E) = 0$ otherwise.

In lines 6–7 of Algorithm 1, $h(a) = h(a_1, \ldots, a_K)$ is the chosen constraint function in Equation 4 and $\partial h / \partial a_i$ is its partial derivative with respect to a_i. For example, to constrain the concentration parameter to equal α, line 6 becomes $h \leftarrow s/\alpha - 1$ and line 7 becomes $J_i \leftarrow 1/\alpha$ for $i \in \{1, \ldots, K\}$. See Section S3 for more details and formulas for handling other constraints.

3 Examples

3.1 Metropolis–Hastings proposals for probabilities

Markov chain Monte Carlo (MCMC) is commonly used for posterior inference in Bayesian models. For models containing a latent vector of probabilities, say $x = (x_1, \ldots, x_K)$, it is

necessary to construct MCMC moves on the probability simplex. Except in cases where the prior on x is Dirichlet and the corresponding likelihood is multinomial, the full conditional distribution of x will rarely have a closed form that can be sampled from. In such cases, a typical approach is to use the Metropolis–Hastings (MH) algorithm to perform an MCMC move that preserves the full conditional distribution of x. A natural choice of MH proposal distribution on the probability simplex is a Dirichlet with mean equal to the current state of x.

However, when the target distribution of x (for instance, the full conditional) places non-negligible mass near the boundary of the simplex, this is highly suboptimal due to the issues illustrated in Figure 1. Specifically, when the current state of x has one or more entries x_i that are near 0, a Dirichlet proposal with mean x will be concentrated either (i) very near x itself or (ii) extremely close to the boundary. In case (i) any moves will be very small, and in case (ii) there will be a very small probability of moving closer to the center of the simplex relative to x. As a result, the MCMC sampler will have difficulty moving around the space efficiently.

The maximum density method can be used to construct MH proposal distributions that yield better MCMC mixing. Specifically, we propose to set the target location c to be the current state of x in the MCMC sampler, set the scale s to a preselected value (for instance, based on pilot runs of the sampler to tune s for good performance), use Algorithm 1 to choose the Dirichlet parameters $a_1, \ldots, a_K$, and then use Dirichlet$(a_1, \ldots, a_K)$ as the MH proposal distribution. The numerator of the MH acceptance ratio can be computed by using the same procedure to define the proposal distribution at the proposed value; see Section S5 for details. This provides better control over the location and scale of the proposals, improving MCMC performance.

To illustrate, we compare several methods for constructing Beta-distributed MH proposals on the unit interval $(0, 1)$. Consider the following four target distributions:

(A) *Uniform:* Beta$(1, 1)$

(B) *Unimodal at zero:* Beta$(1, 1000)$

(C) *Bimodal mixture:* $0.75 \, \mathrm{Beta}(2, 5) + 0.25 \, \mathrm{Beta}(10, 2)$

(D) *Bimodal at zero and one:* Beta$(1/2, 1/2)$.

We compare the performance of the following methods. Letting x denote the current state, consider using an MH proposal consisting of a Beta distribution with:

(I) maximum density for target location x and fixed variance $v = 0.1$,

(II) mean x and fixed concentration parameter $\alpha = 5$,

(III) mean x and fixed variance $v = 0.1$, or

(IV) mean x and standard deviation $\sigma = \min\{x, 1 - x, \sqrt{0.1}\}$.

The choices of $v = 0.1$ and $\alpha = 5$ are based on pilot runs with a range of values to determine choices that perform well; see Figures S4 and S5. Method IV is motivated by the observation that if x is close to 0, then choosing mean x and $\sigma \approx x$ keeps the mass of the distribution relatively near x; likewise for $\sigma \approx 1 - x$ when x is close to 1. Keeping $\sigma \leq \sqrt{0.1}$ avoids issues with non-existence of Beta distributions with large variance, as characterized by Theorem 2. We refer to method IV as using "adaptive variance." Method III appears to be reasonable at first, but is fundamentally flawed since by Theorem 2 there does not always exist a Beta distribution with mean $x \in (0, 1)$ and variance $v = 0.1$. To make Method III well defined, we reject any proposal to a value of x for which a return is impossible due to non-existence of the proposal distribution.

For each target distribution, we run each MCMC sampler for 10,000 iterations after a burn-in of 100 iterations; see Section S5 for a detailed description of the sampler. This is repeated 100 times for each combination of target distribution (A-D) and method (I-IV). To evaluate performance, we consider (i) the autocorrelation function, to quantify mixing performance, and (ii) the Kolmogorov–Smirnov distance between the posterior samples and the target distribution, to verify convergence to the target.

Figure 5 shows the estimated autocorrelation functions (ACFs) for each combination of target distribution (A-D) and method (I-IV). Compared to the ACFs for the mean-based methods, the ACF for our maximum density method decays significantly faster, indicating that the sampler is more efficiently traversing the target distribution. In Figure 5, method III (mean x, variance 0.1) appears to perform reasonably well on target distributions A, C, and D, but this is misleading. In fact, method III is not even converging to the target distribution, as we demonstrate next.

Figure 6 shows the Kolmogorov–Smirnov (KS) distances between the target distribution and the MCMC approximation based on samplers. For each combination of target distribution and method, the figure shows the distribution of KS distances over 100 replicate runs of the MCMC sampler. For the Beta$(1, 1)$ target distribution, all methods except III appear to successfully converge to the target distribution. The reason why method III (mean x, variance 0.1) fails to converge to the target distribution is because by Theorem 2, there exists a Beta with mean x and variance v if and only if $|x - 1/2| < (1/2)\sqrt{1 - 4v}$. Consequently, the sampler for Method III cannot reach any point outside this interval.

For the Beta$(1, 1000)$ target distribution, which is more challenging since it is concentrated near 0, Figure 6 indicates that only methods I (maximum density) and IV (mean x with adaptive variance) successfully converge to the target distribution. Here, even method II (mean x, fixed $\alpha = 5$) fails to converge within the allotted number of MCMC iterations. Since method II is valid, it should eventually converge, but it may take a very large number of iterations. Method III fails again since it is invalid.

Overall, these results show that our maximum density method yields MH samplers that successfully converge to the target distribution (Figure 6) and do so much more rapidly than the mean-based methods (Figure 5).
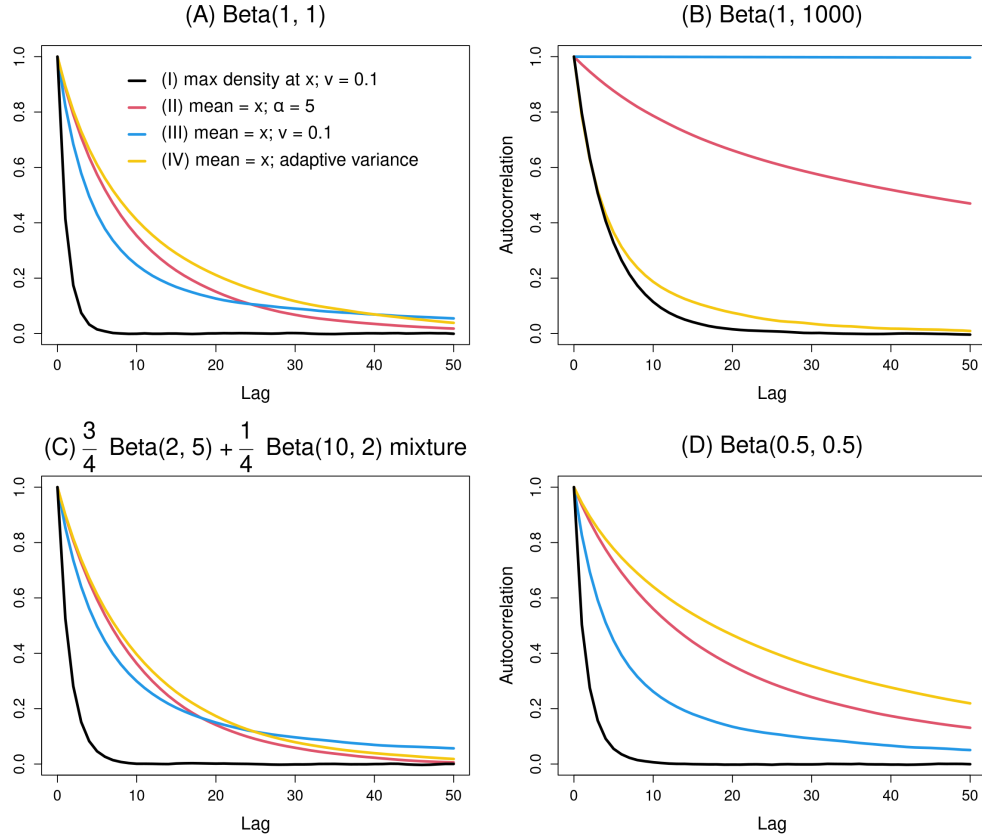
Figure 5: The maximum density method yields MH proposals with improved MCMC mixing compared to the mean method. Each plot shows the autocorrelation functions (ACFs) for one of the four target distributions (A-D). ACFs are shown for each method of proposal distribution specification (I-IV). Maximum density has lower autocorrelation than the mean methods, indicating that the sampler exhibits better performance.

(A) Beta(1, 1)

(I) max density at x; v = 0.1

(II) mean = x; α = 5

(III) mean = x; v = 0.1

(IV) mean = x; adaptive variance

(B) Beta(1, 1000)

(I) max density at x; v = 0.1

(II) mean = x; α = 5

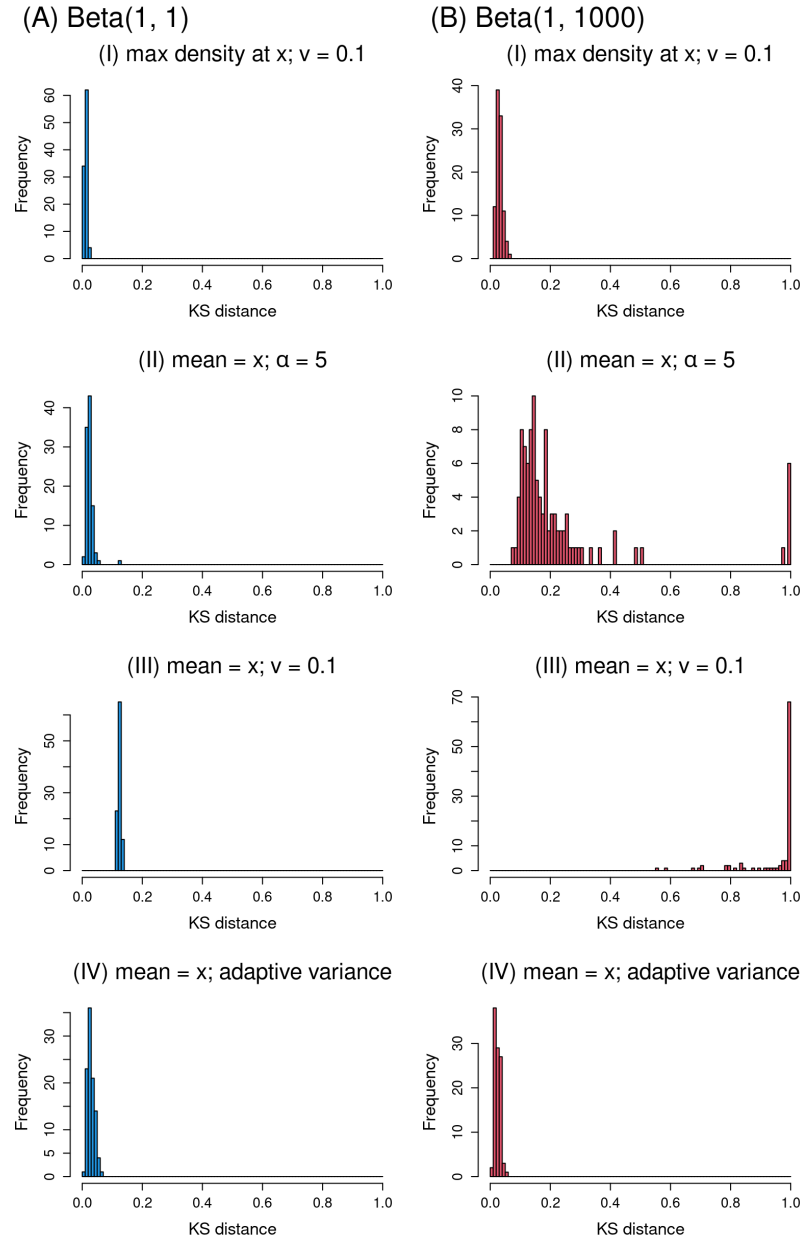### (III) mean = x; v = 0.1

(IV) mean = x; adaptive variance

Figure 6: Histograms of the Kolmogorov–Smirnov distance between the target distribution and the MCMC approximation to it. The blue histograms in the left column are for the Beta(1, 1) target distribution (case A); the red histograms in the right column are for the Beta(1, 1000) target distribution (case B). Samplers using maximum density proposals (method I) or the mean method with adaptive variance (method IV) converge reliably to the target distribution. Method III fails even for simple target distributions. Method II fails when the target distribution puts significant mass near the boundary.

3.2   Bayesian modeling of rare events

Using a good choice of prior is particularly important when performing Bayesian inference for the probability of rare events. Suppose the true probability of the events is on the order of $1/n$ or smaller, where n is the number of observations. In such situations, the prior distribution can have a strong influence on posterior inferences.

Consider a simple Bernoulli model: $Y_1, \ldots, Y_n$ i.i.d. $\sim$ Bernoulli(θ), where Y_i is the observed binary outcome of event i and θ is unknown. A typical choice of prior on θ would be a Beta distribution with mean equal to a location $c \in (0, 1)$ that one expects θ to be near, *a priori*. To illustrate, suppose we expect θ to be around $c = 10^{-3}$, and we use a prior of $\theta \sim \text{Beta}(\alpha c, \alpha(1 - c))$ with concentration parameter $\alpha = 10$, so that the prior mean is c. Now, suppose the true parameter value is θ_0 and the true data generating process is $Y_1, \ldots, Y_n$ i.i.d. $\sim$ Bernoulli(θ_0), where $n = 100$.

One might hope that θ_0 would be a typical value under the posterior distribution of θ. For instance, if the posterior is appropriately quantifying uncertainty about the true value, then 95% credible intervals should contain θ_0 most of the time – ideally, around 95% of the time for correct frequentist calibration. However, Figure 7 (left) shows that this is not the case for the mean method. The coverage of 95% highest posterior density (HPD) intervals is very low for a substantial range of θ_0 values. Even at $\theta_0 = c = 10^{-3}$ (dotted line), where the true parameter equals the mean of the prior distribution, the coverage is only around 10%. The reason is that the prior is concentrated near 0 rather than around c, as illustrated in Figure 1. This also explains why the coverage jumps up to 100% for very small values of θ_0 (less than $\approx 10^{-4.5}$). Thus, the mean method performs worst in the range of θ_0 values where we want it to perform the best (near c).

We propose to instead use our maximum density method to choose the prior on θ. Specifically, we consider using a Beta(a, b) prior with a and b obtained via Equation 2 with target location $c = 10^{-3}$ and concentration parameter $\alpha = 10$. Figure 7 (left) shows that the resulting posteriors are better calibrated, in the sense that θ_0 tends to fall within the 95% HPD interval over a very wide range of θ_0 values, even when the true parameter θ_0 is not particularly close to the prior target location $c = 10^{-3}$. The coverage for both methods drops as θ_0 approaches 1, which makes sense since the prior location of $c = 10^{-3}$ is badly misspecified when θ_0 is close to 1.

Even when the prior is centered at the true parameter, the mean method fails. Suppose the prior is $\theta \sim \text{Beta}(\alpha \theta_0, \alpha(1 - \theta_0))$, so that the prior mean equals the true parameter θ_0. Figure 7 (right) shows that the resulting coverage is low for all values of θ_0 less than around 10^{-2}. Meanwhile, specifying the prior using our maximum density approach with target location $c = \theta_0$ and concentration parameter α yields high coverage for all values of θ_0; see Figure 7 (right). We use $\alpha = 10$ for both methods. Of course, it is unrealistic to make the prior centered at the true value; the point is that even in this ideal situation, the mean method still fails. In contrast, the maximum density method works well—not only in this ideal situation—but also in the realistic situation with a fixed prior that does not depend on θ_0, as shown in Figure 7 (left).
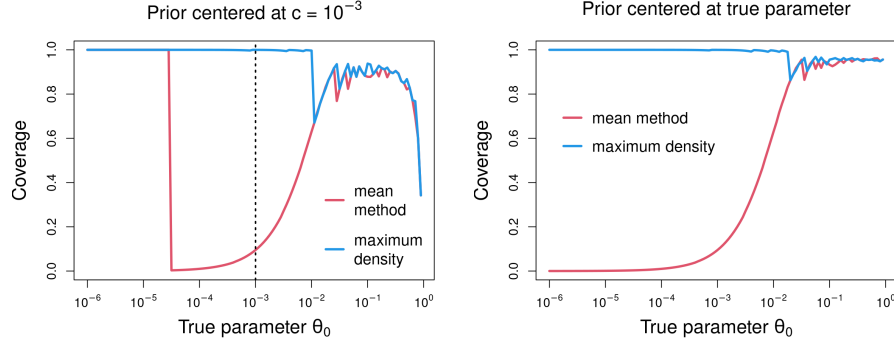
Figure 7: (*Left*) Coverage of 95% highest posterior density intervals as a function of the true parameter θ_0, using Beta priors with concentration $\alpha = 10$ and target location $c = 10^{-3}$ (vertical dotted line). Note: The jagged nature of these lines is due to the discreteness of the distribution, not due to approximation error; the plots show the exact coverage. (*Right*) Same as left, but using Beta priors with target location equal to the true parameter ($c = \theta_0$). The mean method performs very poorly for a wide range of small θ_0 values, whereas our maximum density method performs well across the board.

3.3   Simulating random mutational signatures in cancer genomics

The set of mutations in a cancer genome represents the cumulative effect of numerous mutational processes such as environmental exposures and dysregulated cellular mechanisms. It turns out that each mutational process tends to produce each type of mutation at a relatively constant rate, and these rates can be represented as a probability vector $c = (c_1, \ldots, c_K)$ referred to the corresponding "mutational signature" (Nik-Zainal et al., 2012, 2016; Alexandrov et al., 2013, 2020). Here, c_i represents the rate at which mutation type i occurs for the mutational process under consideration. Usually, one considers the $K = 96$ types of single-base substitution (SBS) mutations; see Figure 8 (left) for examples. The study of mutational signatures has been instrumental in advancing cancer research (Koh et al., 2021; Aguirre et al., 2018; Rubanova et al., 2020).

The Catalogue of Somatic Mutations in Cancer (COSMIC; Alexandrov et al., 2020) provides a curated collection of signatures based on thousands of cancer genomes from a wide range of cancer types. COSMIC signatures are widely used in cancer genome analysis, but it can be important to allow for departures from the COSMIC signatures due to cancer-specific or subject-specific variation (Degasperi et al., 2020; Zou et al., 2021). This variation can be represented by a Dirichlet distribution centered at the signature of interest c (Zito and Miller, 2024). However, if one uses $\text{Dirichlet}(\alpha c_1, \ldots, \alpha c_K)$ where $\alpha > 0$ is the concentration, then the variability around c depends strongly on the sparsity of c. The standard measure of the discrepancy between two mutational signatures, say x and c, is the cosine error,

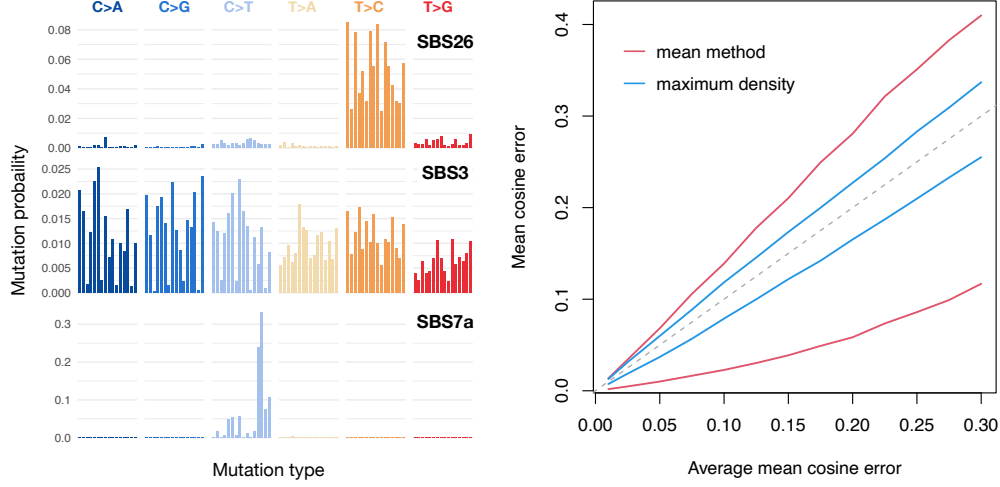$$\text{CosErr}(x, c) = 1 - \frac{x^{\mathsf{T}} c}{\|x\| \|c\|} \tag{5}$$

Figure 8: (*Left*) Three examples of single-base substitution signatures in COSMIC v3.4: SBS26 is associated with defective DNA mismatch repair, SBS3 with homologous recombination deficiencies, and SBS7a with UV light exposure. (*Right*) 25th and 75th percentiles of the empirical distribution of mean cosine errors across all 86 COSMIC v3.4 signatures. For each signature c, we compute the mean cosine error $\mathbb{E}(\mathrm{CosErr}(X, \mathbb{E}(X)))$ when X follows a Dirichlet specified by the mean method as α varies (red lines) or maximum density method as κ varies (blue lines) with target location c. The lines show the percentiles of the empirical distribution of these values across COSMIC signatures c, as a function of the average of this empirical distribution at any given value of α or κ.

for $x, c \in \Delta_K$, where $\|x\| = \sqrt{\sum_i x_i^2}$ is the Euclidean norm. Figure 8 (right) shows that α provides poor control over the mean cosine error between a random signature $X \sim \mathrm{Dirichlet}(\alpha c_1, \ldots, \alpha c_K)$ and its mean c. For any given value of α, the mean cosine error can take a very wide range of values depending on the signature c. Consequently, α does not represent the scale of variability around COSMIC signatures in a consistent way across signatures, when using the mean method.

We propose to instead use our maximum density method to specify Dirichlet distributions for representing variability around mutational signatures. Specifically, for a given COSMIC signature c and a desired mean cosine error κ, we aim to maximize $\mathrm{Dirichlet}(c \mid a_1, \ldots, a_K)$ subject to the constraint that $\mathbb{E}(\mathrm{CosErr}(X, \mathbb{E}(X))) = \kappa$, where $X \sim \mathrm{Dirichlet}(a_1, \ldots, a_K)$. Since the mean cosine error $\mathbb{E}(\mathrm{CosErr}(X, \mathbb{E}(X)))$ is not mathematically tractable, we use the expected value of a second-order Taylor approximation to the cosine error; see Section S6 for details. We then use Algorithm 1 to maximize $\mathrm{Dirichlet}(c \mid a_1, \ldots, a_K)$ subject to constraining this approximation to the mean cosine error; see Section S3 for the precise formulas we use for the constraint function $h(a)$ and its Jacobian matrix J. We find that this algorithm converges reliably in all of the settings we have tried.

Figure 8 (right) shows that our maximum density method provides much better control over the mean cosine error. More precisely, the distribution of mean cosine errors across COSMIC signatures is much tighter, indicating that mean cosine error is being more effectively controlled. This is shown by having a smaller gap between 25th and 75th percentiles of the distribution of mean cosine errors across COSMIC signatures. This demonstrates that the maximum density method provides better control over the scale of Dirichlet distributions in this real world application. This example also illustrates the flexibility of the maximum density method, since one can choose a measure of scale that is relevant for the application at hand.

4   Discussion

The usual way of specifying Beta and Dirichlet distributions, by setting the mean equal to the target location, is prone to exhibit pathological behavior near the boundary. This underappreciated problem may lead to poorly performing MCMC algorithms or unintentionally strong priors that induce significant bias. We introduce a novel approach that provides better control over the location and scale of these distributions.

This issue—and our proposed solution—may be relevant in a wide range of applications in which Dirichlet distributions play a central role, including mutational signatures analysis (Zito and Miller, 2024), microbiome analysis of compositional count matrices (Chen and Li, 2013), species abundance data (Bersson and Hoff, 2024), document analysis with topic models (Blei et al., 2003), population structure analysis with admixture models (Pritchard et al., 2000), and applications of mixture models (Miller and Harrison, 2018).

Another application of particular interest is variable selection with global-local shrinkage priors. Dirichlet–Laplace priors (Bhattacharya et al., 2015; Zhang and Bondell, 2018) are used to model the local shrinkage parameters of the mean of each observation using a Dirichlet distribution, while the global variance is assigned a gamma prior. With current methods, inference is only feasible using Gibbs sampling for very specific choices of the gamma hyperparameters. Our maximum density method may facilitate the design of efficient proposals for the local parameters when they are close to zero.

Finally, the maximum density method could be applied to other distributions, beyond the Beta and Dirichlet. The technique of maximizing the density at a target location, subject to a scale constraint, makes sense for many families of distributions.

Acknowledgments

Supplementary Material

The Supplementary Material includes further results and analyses. Code implementing the method and examples in the paper is publicly available via at https://github.com/casxue/ImprovedDirichlet.

References

Aguirre, A. J., Nowak, J. A., Camarda, N. D., et al. (2018). "Real-time Genomic Characterization of Advanced Pancreatic Cancer to Enable Precision Medicine." *Cancer Discovery*, 8(9): 1096–1111.
URL https://doi.org/10.1158/2159-8290.CD-18-0275 15

Alexandrov, L. B., Kim, J., Haradhvala, N. J., et al. (2020). "The repertoire of mutational signatures in human cancer." *Nature*, 578(7793): 94–101. 15

Alexandrov, L. B., Nik-Zainal, S., Wedge, D. C., et al. (2013). "Signatures of mutational processes in human cancer." *Nature*, 500(7463): 415–421.
URL https://doi.org/10.1038/nature12477 15

Bersson, E. and Hoff, P. D. (2024). "Frequentist Prediction Sets for Species Abundance using Indirect Information." *Journal of the Royal Statistical Society Series A: Statistics in Society*, qnae096.
URL https://doi.org/10.1093/jrsssa/qnae096 17

Bhattacharya, A., Pati, D., Pillai, N. S., and Dunson, D. B. (2015). "Dirichlet–Laplace Priors for Optimal Shrinkage." *Journal of the American Statistical Association*, 110(512): 1479–1490. PMID: 27019543.
URL https://doi.org/10.1080/01621459.2014.960967 2, 17

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). "Latent Dirichlet allocation." *Journal of Machine Learning Research*, 3: 993–1022. 2, 17

Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press. 8

Chen, J. and Li, H. (2013). "Variable selection for sparse Dirichlet-multinomial regression with an application to microbiome data analysis." *The Annals of Applied Statistics*, 7(1). 17

Degasperi, A., Amarante, T. D., Czarnecki, J., Shooter, S., Zou, X., Glodzik, D., Morganella, S., Nanda, A. S., Badja, C., Koh, G., Momen, S. E., Georgakopoulos-Soares, I., Dias, J. a. M. L., Young, J., Memari, Y., Davies, H., and Nik-Zainal, S. (2020). "A practical framework and online tool for mutational signature analyses show inter-tissue variation and driver dependencies." *Nature Cancer*, 1: 249–263. 15

Koh, G., Degasperi, A., Zou, X., Momen, S., and Nik-Zainal, S. (2021). "Mutational signatures: Emerging concepts, caveats and clinical applications." *Nature Reviews Cancer*, 21(10): 619–637.
URL https://doi.org/10.1038/s41568-021-00377-7 15

Miller, J. W. and Harrison, M. T. (2018). "Mixture models with a prior on the number of components." *Journal of the American Statistical Association*, 113(521): 340–356.
URL https://doi.org/10.1080/01621459.2016.1255636 2, 17

Nik-Zainal, S., Alexandrov, L. B., Wedge, D. C., et al. (2012). "Mutational Processes Molding the Genomes of 21 Breast Cancers." *Cell*, 149(5): 979–993.
URL https://doi.org/10.1016/j.cell.2012.04.024 15

Nik-Zainal, S., Davies, H., Staaf, J., et al. (2016). "Landscape of somatic mutations in 560 breast cancer whole-genome sequences." *Nature*, 534(7605): 47–54.
URL https://doi.org/10.1038/nature17676 15

Pritchard, J. K., Stephens, M., and Donnelly, P. (2000). "Inference of population structure using multilocus genotype data." *Genetics*, 155(2): 945–959. 2, 17

Richardson, S. and Green, P. J. (1997). "On Bayesian analysis of mixtures with an unknown number of components (with discussion)." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 59(4): 731–792.
URL https://doi.org/10.1111/1467-9868.00146 2

Rousseau, J. and Mengersen, K. (2011). "Asymptotic behaviour of the posterior distribution in overfitted mixture models." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(5): 689–710.
URL https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9868.2011.00781.x 2

Rubanova, Y., Shi, R., Harrigan, C. F., et al. (2020). "Reconstructing evolutionary trajectories of mutation signature activities in cancer using TrackSig." *Nature Communications*, 11(1): 731.
URL https://doi.org/10.1038/s41467-020-14352-7 15

Zhang, Y. and Bondell, H. D. (2018). "Variable Selection via Penalized Credible Regions with Dirichlet–Laplace Global-Local Shrinkage Priors." *Bayesian Analysis*, 13(3): 823 – 844.
URL https://doi.org/10.1214/17-BA1076 2, 17

Zito, A. and Miller, J. W. (2024). "Compressive Bayesian non-negative matrix factorization for mutational signatures analysis."
URL https://arxiv.org/abs/2404.10974 2, 15, 17

Zou, X., Koh, G. C. C., Nanda, A. S., Degasperi, A., Urgo, K., Roumeliotis, T. I., Agu, C. A., Badja, C., Momen, S., Young, J., Amarante, T. D., Side, L., Brice, G., Perez-Alonso, V., Rueda, D., Gomez, C., Bushell, W., Harris, R., Choudhary, J. S., Consortium, G. E. R., Jiricny, J., Skarnes, W. C., and Nik-Zainal, S. (2021). "A systematic CRISPR screen defines mutational mechanisms underpinning signatures caused by replication errors and endogenous DNA damage." *Nature Cancer*, 2: 643–657. 15

Supplementary Material for
"Improved control of Dirichlet location and scale near the boundary"

S1 Proofs

This section provides the proofs of the results in the paper.

Proof of Theorem 1. Let $\Theta = (0,\infty)^2 \subseteq \mathbb{R}^2$. Define $f : \Theta \to \mathbb{R}$ by $f(a,b) = -\log \text{Beta}(c \mid a,b)$ and define $S = \{(a,b) \in \Theta : V(a,b) = v\}$. In terms of f and S, the optimization problem in Theorem 1 is

$$\arg\min_{a,b>0} f(a,b) \text{ subject to } (a,b) \in S. \tag{6}$$

First, S is nonempty since by Theorem 2 there exists a Beta distribution with variance v whenever $0 < v < 1/4$. Furthermore, S is bounded, since

$$\min\{1/a, 1/b\} > \frac{1}{a+b+1} \geq \frac{a}{a+b}\frac{b}{a+b}\frac{1}{a+b+1} = V(a,b) = v > 0$$

for all $(a,b) \in S$, and therefore $\max\{a,b\} < 1/v$ for all $(a,b) \in S$. Now, for all $a,b \in (0,1]$, we have $\text{B}(a,b) \geq (a+b-ab)/(ab) = 1/a + 1/b - 1$, by Ivády (2016); also see Zhao and Wang (2023). Therefore, for $a,b \in (0,1)$,

$$f(a,b) = -(a-1)\log(c) - (b-1)\log(1-c) + \log \text{B}(a,b)$$
$$\geq \log(c) + \log(1-c) + \log(1/a + 1/b - 1). \tag{7}$$

Let (a',b') be any point in S. By Equation 7, we can choose $\varepsilon \in (0,1)$ small enough that $f(a,b) > f(a',b') + 1$ for all (a,b) such that $a < \varepsilon$ or $b < \epsilon$. Define $A = [\varepsilon,\infty)^2$. It follows that $(a',b') \in A \cap S$ and $f(a',b') < \inf\{f(a,b) : (a,b) \in \Theta \setminus A\}$. Thus, no point of $\Theta \setminus A$ can be a solution, so minimizing over S is equivalent to minimizing over $A \cap S$. To complete the proof, we just need to show that the minimum of f over $A \cap S$ is attained at some point; any such point will be a solution to Equation 6.

Let $V|_A$ denote the restriction of V to A. Then $V|_A$ is a continuous function, so the pre-image of $\{v\}$ is closed, that is, $\{(a,b) \in A : V(a,b) = v\} = A \cap S$ is a closed subset of A, and hence, a closed subset of $\mathbb{R}^2$. Since $A \cap S$ is also bounded (as a subset of the bounded set S), it is a compact set. Therefore, the continuous function f attains its minimum on $A \cap S$. $\square$

Theorem S1. *For all $c \in (0,1)$ and $\alpha \in (0,\infty)$, there exists a finite solution to:*

$$\arg\max_{a,b>0} \text{Beta}(c \mid a,b) \text{ subject to } a+b = \alpha.$$

Proof. The proof is essentially the same as the proof of Theorem 1, except that $S = \{(a,b) \in \Theta : a+b = \alpha\}$. All that is needed is to recognize that S is a nonempty, bounded set, and the rest of the proof is the same. $\square$

S1

Proof of Theorem 2. Reparametrize the $\text{Beta}(a,b)$ distribution in terms of the mean $u = a/(a+b)$ and concentration $\alpha = a+b$, so that $a = \alpha u$ and $b = \alpha(1-u)$. The variance v of $\text{Beta}(a,b)$ is related to u and α via

$$v = \frac{ab}{(a+b)^2(a+b+1)} = \frac{u(1-u)}{\alpha+1}. \tag{8}$$

Thus, solving for α as a function of u and v yields that $\alpha = u(1-u)/v - 1$. For any $v > 0$, define

$$f_v(u) = \frac{u(1-u)}{v} - 1.$$

Since α must be positive to be the concentration of a Beta distribution, only a subset of u and v values can be the mean and variance of a Beta distribution. More precisely, for any $u \in (0,1)$ and $v > 0$, there is a Beta distribution with mean u and variance v if and only if $f_v(u) > 0$.

Note that $u \mapsto u(1-u)$ is a quadratic function with a maximum of $u(1-u) = 1/4$ at $u = 1/2$. Along with Equation 8, this implies that the variance must satisfy $0 < v < 1/4$ since $\alpha > 0$. To find the feasible range of u values for any given v, we set

$$0 = f_v(u) = \frac{u(1-u)}{v} - 1 = -u^2/v + u/v - 1$$

and solve using the quadratic formula to find

$$u_* = \frac{-1/v + \sqrt{1/v^2 - 4/v}}{-2/v} = 1/2 - (1/2)\sqrt{1-4v} \tag{9}$$

$$u^* = \frac{-1/v - \sqrt{1/v^2 - 4/v}}{-2/v} = 1/2 + (1/2)\sqrt{1-4v}. \tag{10}$$

Thus, for any $v \in (0, 1/4)$, we have $f_v(u) > 0$ if and only if $u_* < u < u^*$, where u_* and u^* given by Equations 9 and 10. Therefore, there is a Beta distribution with mean u and variance v if and only if $0 < v < 1/4$ and $|u - 1/2| < (1/2)\sqrt{1-4v}$. $\quad\square$

S2 Distance between logit-transformed values

In this section, we derive the density of the distance between a Beta random variable and a given point, after logit transformation. Furthermore, we describe how to compute this density in a numerically stable way.

Fix $a > 0$, $b > 0$, and $c \in (0,1)$. Let $X \sim \text{Beta}(a,b)$ and define

$$Y = \big|\text{logit}(X) - \text{logit}(c)\big|,$$

where $\text{logit}(x) = \log(x/(1-x))$ for $x \in (0,1)$. Let $f_X(x)$ denote the probability density function of X, namely,

$$f_X(x) = \frac{1}{\text{B}(a,b)}x^{a-1}(1-x)^{b-1}\mathbb{1}(0 < x < 1).$$

In Section S2.2, we show that the probability density function of Y is

$$f_Y(y) = f_X\left(\frac{1}{1+e^{y-\ell}}\right)\frac{e^{y-\ell}}{(1+e^{y-\ell})^2} + f_X\left(\frac{1}{1+e^{-y-\ell}}\right)\frac{e^{-y-\ell}}{(1+e^{-y-\ell})^2} \qquad (11)$$

for $y \in (0, \infty)$, and $f_Y(y) = 0$ otherwise.

S2.1 Numerically stable computation of the density of Y

Computing $f_Y(y)$ requires careful handling of the exponentials in order to avoid numerical underflow or overflow issues. The first point to note is that one should work with logarithms rather than the values themselves. Specifically, compute the log of each term in Equation 11, and combine them using the "logsumexp" trick to obtain $\log f_Y(y)$, that is, $\log(\exp(r) + \exp(s)) = \log(\exp(r-m) + \exp(s-m)) + m$ where $m = \max\{r, s\}$. The next point is that computing $\log(1 + e^x)$ is prone to numerical issues: When $x \ll 0$, it rounds off to 0, whereas when $x \gg 0$, it overflows to ∞. This can be fixed by (i) using a `log1p` function, which computes $\log(1 + t)$ in a numerically accurate way for small t, along with (ii) a conditional to avoid cases where e^x would overflow, for instance:

$$\log(1 + e^x) \approx \begin{cases} \texttt{log1p}(\exp(x)) & \text{if } x < 0 \\ \texttt{log1p}(\exp(-x)) + x & \text{if } x \geq 0. \end{cases}$$

S2.2 Derivation of the density of Y

To derive Equation 11, we use Jacobi's formula for transformation of continuous random random variables to derive the probability density function of Y, as follows. Let $A_0 = \{c\}$, $A_1 = (0, c)$, and $A_2 = (c, 1)$, noting that $\{A_0, A_1, A_2\}$ is a partition of $\mathcal{X} := \{x : f_X(x) > 0\} = (0, 1)$ such that $P(X \in A_0) = 0$. Define $g(x) = |\text{logit}(x) - \text{logit}(c)|$, and observe that g is strictly monotone on each of A_1 and A_2, separately. Furthermore, letting g_1 and g_2 be the restrictions of g to A_1 and A_2, respectively, it holds that the inverse g_i^{-1} exists and is continuously differentiable on $g(A_i) = \{g(x) : x \in A_i\}$, for $i = 1, 2$. Thus, by Jacobi's transformation formula (Casella and Berger, 2024; Jacod and Protter, 2012), the probability density of Y is

$$f_Y(y) = \sum_{i=1}^{2} f_X(g_i^{-1}(y))\left|\frac{d}{dy}g_i^{-1}(y)\right|\mathbb{1}(y \in g(A_i)). \qquad (12)$$

To use this formula, we just need to derive g_i^{-1} and its derivative for $i = 1, 2$. To this end, observe that

$$g_1(x) = \log\left(\frac{c}{1-c}\right) - \log\left(\frac{x}{1-x}\right) \qquad\qquad x \in (0, c),$$
$$g_2(x) = \log\left(\frac{x}{1-x}\right) - \log\left(\frac{c}{1-c}\right) \qquad\qquad x \in (c, 1).$$

Define $\ell = \log(c/(1-c))$ to simplify the notation. Solving for the inverses, we obtain

$$g_1^{-1}(y) = \frac{1}{1+e^{y-\ell}} \qquad\qquad\qquad y \in (0, \infty), \qquad (13)$$

S3

$$g_2^{-1}(y) = \frac{1}{1 + e^{-y-\ell}} \qquad\qquad y \in (0, \infty), \qquad (14)$$

and differentiating, we have

$$\frac{d}{dy} g_1^{-1}(y) = \frac{-e^{y-\ell}}{(1 + e^{y-\ell})^2} \qquad\qquad y \in (0, \infty), \qquad (15)$$

$$\frac{d}{dy} g_2^{-1}(y) = \frac{e^{-y-\ell}}{(1 + e^{-y-\ell})^2} \qquad\qquad y \in (0, \infty). \qquad (16)$$

Plugging Equations 13 to 16 into Equation 12 yields

$$f_Y(y) = f_X\left(\frac{1}{1 + e^{y-\ell}}\right)\frac{e^{y-\ell}}{(1 + e^{y-\ell})^2} + f_X\left(\frac{1}{1 + e^{-y-\ell}}\right)\frac{e^{-y-\ell}}{(1 + e^{-y-\ell})^2} \qquad (17)$$

for $y \in (0, \infty)$, and $f_Y(y) = 0$ otherwise.

S3   Derivation of the optimization algorithm

In this section, we derive Algorithm 1 based on Newton's method with equality constraints (Boyd and Vandenberghe, 2004). This technique provides a second-order optimization algorithm for solving problems of the form

$$\arg\min_x f(x) \text{ subject to } h(x) = 0,$$

where f is a strictly convex, twice-differentiable function and h is differentiable. Let $g(x) = \nabla f(x) = (\partial f/\partial x_i)$ denote the gradient of f, let $H(x) = \nabla^2 f(x) = (\partial^2 f/\partial x_i \partial x_j)$ denote the Hessian matrix of f, and let $J(x) = (\partial h_i/\partial x_j)$ denote the Jacobian matrix of h. After initializing x to an appropriate value, each iteration proceeds by updating

$$x \leftarrow x + \delta \qquad (18)$$

where the vector δ is defined by solving the linear system

$$\begin{bmatrix} H(x) & J(x)^{\mathsf{T}} \\ J(x) & 0 \end{bmatrix} \begin{bmatrix} \delta \\ \lambda \end{bmatrix} = \begin{bmatrix} -g(x) \\ -h(x) \end{bmatrix}. \qquad (19)$$

Here, λ is a vector of multipliers that we will not use in our algorithm.

S3.1   Applying Newton's method to the maximum density method

To apply this to implement the maximum density method as in Equation 4, we define $x = a = (a_1, \ldots, a_K)$ and

$$f(a) = -\log \text{Dirichlet}(c \mid a) = \sum_{i=1}^{K} \log \Gamma(a_i) - \log \Gamma\left(\sum_{i=1}^{K} a_i\right) - \sum_{i=1}^{K} (a_i - 1) \log(c_i).$$

Then f is smooth and strictly convex on $(0, \infty)^K$ by Miller and Harrison (2014, Proposition 19). For the constraint in the Dirichlet case, we consider two options: (i) fixed concentration α, and (ii) fixed mean cosine error κ. For the special case of Beta distributions, see Section S3.3. We show how to handle each of these constraints below.

The gradient and Hessian of $f(a)$ are obtained by differentiating:

$$\frac{\partial f}{\partial a_i}(a) = \psi(a_i) - \psi\left(\sum_i a_i\right) - \log(c_i),$$

$$\frac{\partial^2 f}{\partial a_i \partial a_j}(a) = \psi'(a_i)\mathbb{1}(i = j) - \psi'\left(\sum_i a_i\right).$$

Thus, the functions g and H take the following forms, where $s = \sum_i a_i$:

$$g(a) = \nabla f(a) = \left[\psi(a_i) - \psi(s) - \log(c_i)\right] \in \mathbb{R}^{K \times 1}$$

$$H(a) = \nabla^2 f(a) = \left[\psi'(a_i)\mathbb{1}(i = j) - \psi'(s)\right] \in \mathbb{R}^{K \times K}$$

where $\psi(x) = (d/dx)\log\Gamma(x)$ is the digamma function and $\psi'(x) = (d/dx)\psi(x)$ is the trigamma function.

Constraining the concentration parameter

First, we consider constraining the concentration parameter α, in which case we define

$$h(a) = \frac{\sum_{i=1}^K a_i}{\alpha} - 1. \tag{20}$$

Note that $h(a) = 0$ if and only if $\alpha = \sum_{i=1}^K a_i$. In this case, h is a linear function of a, which is conducive for convergence since we are optimizing a strictly convex function over a convex set. The rationale for defining h using the ratio $(\sum_i a_i)/\alpha$ rather than the difference $(\sum_i a_i) - \alpha$ is so that the convergence tolerance can be relatively invariant to the magnitude of α; when using the ratio, the number of significant digits is what matters. The Jacobian matrix is obtained by differentiating Equation 20,

$$\frac{\partial h}{\partial a_i} = \frac{\partial}{\partial a_i}\left(\frac{\sum_{i=1}^K a_i}{\alpha} - 1\right) = \frac{1}{\alpha},$$

and thus,

$$J(a) = \begin{bmatrix} 1/\alpha & \cdots & 1/\alpha \end{bmatrix}.$$

Constraining the mean cosine error

For the case of mean cosine error, we use the following approximation to the mean cosine error between $X \sim \text{Dirichlet}(a_1, \dots, a_K)$ and $\mathbb{E}(X)$,

$$\mathbb{E}(\text{CosErr}(X, \mathbb{E}(X))) \approx \frac{\sum_i a_i}{2(1 + \sum_i a_i)(\sum_i a_i^2)}\left(\sum_i a_i - \frac{\sum_i a_i^3}{\sum_i a_i^2}\right) \tag{21}$$

S5

where each sum is over $i = 1, \ldots, K$; see Section S6 for the derivation. For the constraint function h, we work with the logarithm, defining

$$h(a) = -\log(2) + \log(\textstyle\sum_i a_i) - \log(1 + \textstyle\sum_i a_i) - \log(\textstyle\sum_i a_i^2)$$
$$+ \log\left(\textstyle\sum_i a_i - \frac{\sum_i a_i^3}{\sum_i a_i^2}\right) - \log(\kappa)$$

where $\kappa > 0$ is the desired mean cosine error. The rationale for defining h using the logarithm is so that the derivatives take mathematically simple forms. Additionally, this expresses the constraint in terms of the ratio rather than the difference, which is advantageous for the same reasons discussed in the concentration parameter case. Differentiating, we have

$$\frac{\partial h}{\partial a_j} = \frac{1}{\sum_i a_i} - \frac{1}{1 + \sum_i a_i} - \frac{2a_j}{\sum_i a_i^2} + \frac{1 - \left(3a_j^2(\sum_i a_i^2) - 2a_j(\sum_i a_i^3)\right)/(\sum_i a_i^2)^2}{(\sum_i a_i) - (\sum_i a_i^3)/(\sum_i a_i^2)}.$$

Defining $s_k = \sum_i a_i^k$ for $k \in \{1, 2, 3\}$, the formulas can be expressed more compactly as

$$h(a) = -\log(2) + \log(s_1) - \log(1 + s_1) - \log(s_2) + \log(s_1 - s_3/s_2) - \log(\kappa)$$
$$\frac{\partial h}{\partial a_j} = \frac{1}{s_1} - \frac{1}{1 + s_1} - \frac{2a_j}{s_2} + \frac{1 - (3a_j^2 s_2 - 2a_j s_3)/s_2^2}{s_1 - s_3/s_2}.$$

Even though h is a nonlinear function of a, we find that the algorithm still successfully converges once some implementation details have been addressed; we discuss this next.

S3.2 Implementation details

To implement the basic version of the algorithm described above, at each iteration we would compute the expressions for $g(a)$, $H(a)$, and $J(a)$, plug them into Equation 19, use a linear solver to obtain δ, and update a as in Equation 18. In practice, we modify the algorithm to improve its numerical stability. First, we modify Equation 18 to use a step size of $\rho \in (0, 1)$, so that the update is

$$a \leftarrow a + \rho\,\delta. \tag{22}$$

Second, we enforce the boundary constraint that $a_1, \ldots, a_K$ must be positive numbers as follows: for each $i = 1, \ldots, K$, if a_i would be less than or equal to zero after the update in Equation 22, then we instead update $a_i \leftarrow a_i/2$ for that i. This maintains positivity, but still moves a_i in the direction of the full Newton step.

We begin with an initial step size of $\rho = 1/2$, and we initialize the algorithm at $a = 10(c + 1)/2$, that is, $a_i = 10(c_i + 1)/2$ for $i = 1, \ldots, K$. As a stopping criterion, we halt the algorithm after either (i) a maximum number of iterations `maxiter` has been reached (default: `maxiter = 100`), or (ii) $|h(a)| + \sum_{i=1}^{K} |a_i/a_i' - 1| < \texttt{tol}$, where $a_1', \ldots, a_K'$ are the values of $a_1, \ldots, a_K$ from the previous iteration and `tol > 0` is a convergence tolerance (default: $\texttt{tol} = 10^{-8}$).

If the algorithm reaches the maximum number of iterations without converging to within the specified tolerance, we restart the algorithm with $\rho \leftarrow \rho/5$ and $\texttt{maxiter} \leftarrow 5\,\texttt{maxiter}$. If the algorithm still fails after 5 adaptive restarts of this form, we stop and report that the procedure has failed. We find that this adaptive procedure reliably yields successful convergence for a wide range of values.

S3.3 Algorithm in the special case of Beta distributions

For concreteness, Algorithm S1 provides a version of Algorithm 1 that is specialized to Beta distributions. To constrain the concentration to some value α, lines 8–10 of Algorithm S1 become $h \leftarrow (a+b)/\alpha - 1$, $J_1 \leftarrow 1/\alpha$, and $J_2 \leftarrow 1/\alpha$. Meanwhile, to constrain the variance to some value $v \in (0, 1/4)$, lines 8–10 become

$$h \leftarrow \log(a) + \log(b) - 2\log(a+b) - \log(a+b+1) - \log(v)$$
$$J_1 \leftarrow 1/a - 2/(a+b) - 1/(a+b+1)$$
$$J_2 \leftarrow 1/b - 2/(a+b) - 1/(a+b+1).$$

S4 Additional details on the median method

As discussed in Section 2, an alternative approach is to choose a Beta distribution with median equal to the target location c, and with either a given variance v or a given concentration parameter α. Figure S1 shows the plots of CDFs and percentiles for the median method with $c \in \{0.001, 0.2\}$, for a range of α values (compare with Figures 1 and 2). Like maximum density, the median method provides better control over the location and scale of Beta distributions than the mean method. The median method has the further advantage of directly centering the distribution at c (in terms of the median) by construction.

However, as shown in Figure S2, the median method puts somewhat less mass near the target location c, compared to the maximum density method. The objective function for median method is also more complicated to optimize, compared to maximizing the density as in our proposed method. Furthermore, importantly, the median method is not as straightforward to extend to the general Dirichlet case.

On the Metropolis–Hastings example in Section 3.1, the median method performs similarly to the maximum density method as a technique for constructing MH proposal distributions. Figure S3 shows the autocorrelation functions for the mean, median, and maximum density methods for a range of parameter settings. To facilitate comparison with the other methods, we consider proposals with median c and (i) fixed concentration $\alpha = 5$, (ii) fixed variance $v = 0.1$, and (iii) adaptive variance, in other words, $\sigma = \min\{x, 1-x, \sqrt{0.1}\}$.

Algorithm S1   Maximum density method for Beta distributions

Input: target location $c \in (0,1)$, step size $\rho \in (0,1]$, maximum number of iterations `maxiter` > 0, and convergence tolerance `tol` > 0.

1: $a \leftarrow c$ and $b \leftarrow 1 - c$ ▷ Initialization
2: **for** $i = 1, \ldots, $ `maxiter` **do**
3:      $g_1 \leftarrow -\log(c) - \psi(a+b) + \psi(a)$ ▷ Compute gradient
4:      $g_2 \leftarrow -\log(1-c) - \psi(a+b) + \psi(b)$
5:      $H_{11} \leftarrow -\psi'(a+b) + \psi'(a)$ ▷ Compute Hessian matrix
6:      $H_{22} \leftarrow -\psi'(a+b) + \psi'(b)$
7:      $H_{12} \leftarrow -\psi'(a+b)$
8:      $h \leftarrow h(a,b)$ ▷ Constraint
9:      $J_1 \leftarrow \partial h / \partial a$ ▷ Jacobian of constraint
10:     $J_2 \leftarrow \partial h / \partial b$
11:     Solve for δ_1 and δ_2 in the linear system: ▷ Compute Newton step

$$
\begin{bmatrix} H_{11} & H_{12} & J_1 \\ H_{12} & H_{22} & J_2 \\ J_1 & J_2 & 0 \end{bmatrix} \begin{bmatrix} \delta_1 \\ \delta_2 \\ \lambda \end{bmatrix} = \begin{bmatrix} -g_1 \\ -g_2 \\ -h \end{bmatrix}
$$

12:     $a' \leftarrow a$ and $b' \leftarrow b$ ▷ Store current values
13:     $a \leftarrow a' + \rho\,\delta_1$ and $b \leftarrow b' + \rho\,\delta_2$ ▷ Update values
14:     **if** $a \leq 0$ **then** $a \leftarrow a'/2$ ▷ Enforce boundary constraints
15:     **if** $b \leq 0$ **then** $b \leftarrow b'/2$
16:     **if** $|h| + |a/a' - 1| + |b/b' - 1| < $ `tol` **then**   ▷ Check for convergence
17:         **output** a and b ▷ Return output
18:     **end if**
19: **end for**
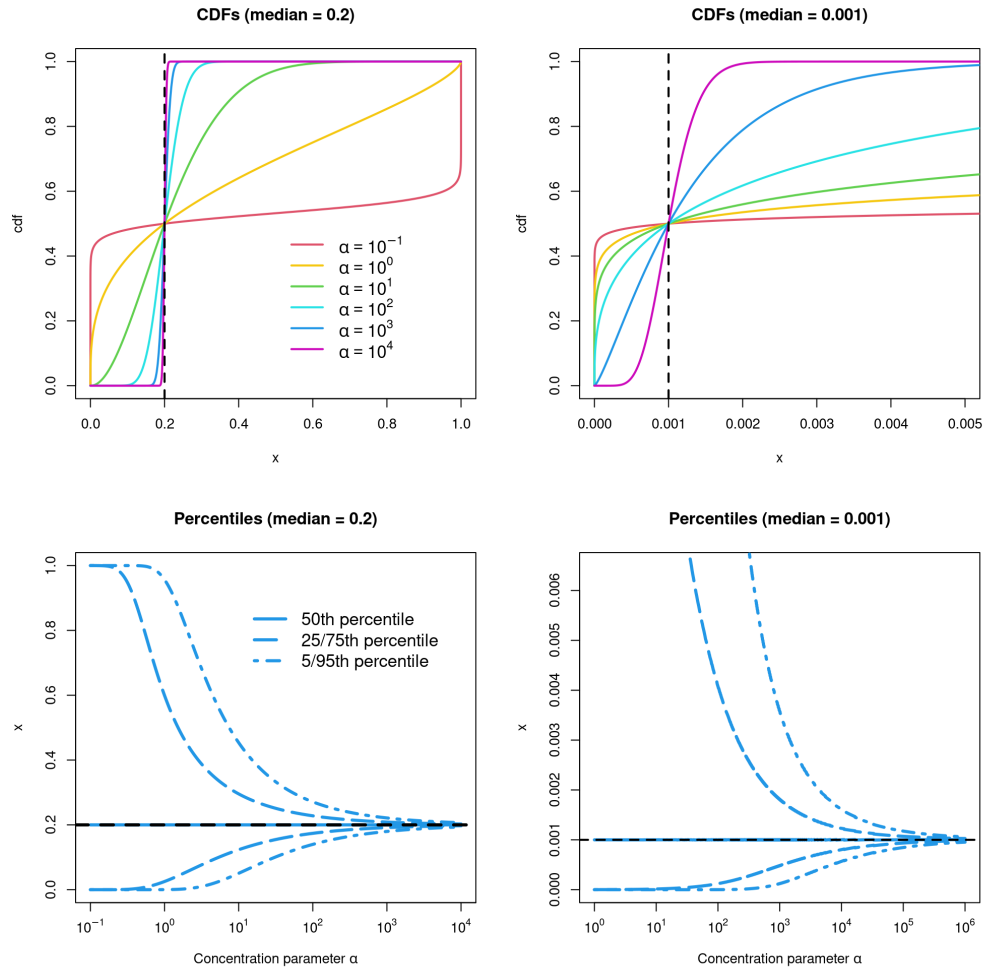**Output:** Beta parameters a and b.

Figure S1: The median method performs similarly to our maximum density method in terms of providing improved control over the scale. See the captions of Figures 1 and 2 for description of the plots.
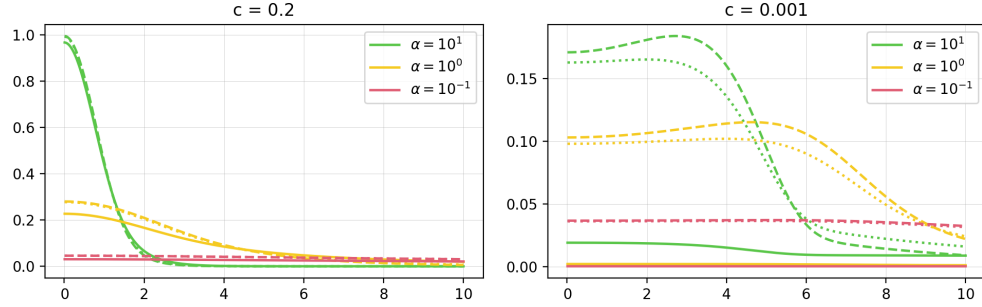
Figure S2: The maximum density method (dashed lines) puts more probability mass near the target location, compared to both the mean method (solid lines) and median method (dotted lines). See the caption of Figure 3 for description of the plots. The concentration α is the same for all three methods. The median and maximum density curves are nearly indistinguishable in the case of $c = 0.2$.

S5 Metropolis–Hastings example details

This section provides more detail on the MCMC example in Section 3.1. For $x \in (0, 1)$, define a_x and b_x to be the values of a and b obtained via a given method. For example, for method II (mean at current value x and concentration parameter α), we have $a_x = \alpha x$ and $b_x = \alpha(1 - x)$. For method I (maximum density with target location x and fixed variance v), a_x and b_x are the solution to Equation 1 with $c = x$ and $h(a, b) = V(a, b)/v - 1$.

The Metropolis–Hastings algorithm then proceeds as follows. Let $\pi(x)$ denote the density of the target distribution, and suppose the current state is x. Sample a proposed value $x' \sim \text{Beta}(a_x, b_x)$. Compute the acceptance probability

$$ p = \min \left\{ 1, \ \frac{\pi(x') \, \text{Beta}(x \mid a_{x'}, b_{x'})}{\pi(x) \, \text{Beta}(x' \mid a_x, b_x)} \right\}. $$

With probability p, accept the proposal (so the state becomes x'), and otherwise, reject the proposal (so the state remains x). We initialize the sampler by setting the initial state to be 0.25, and then each iteration of the MCMC algorithm consists of one MH move as described above.

Figures S4 and S5 show the results of tuning the parameters of each proposal distribution to optimize their performance in the MH example in Section 3.1. The plots show the autocorrelation functions (ACFs) for a range of values of α (for method II) and v (for method I).
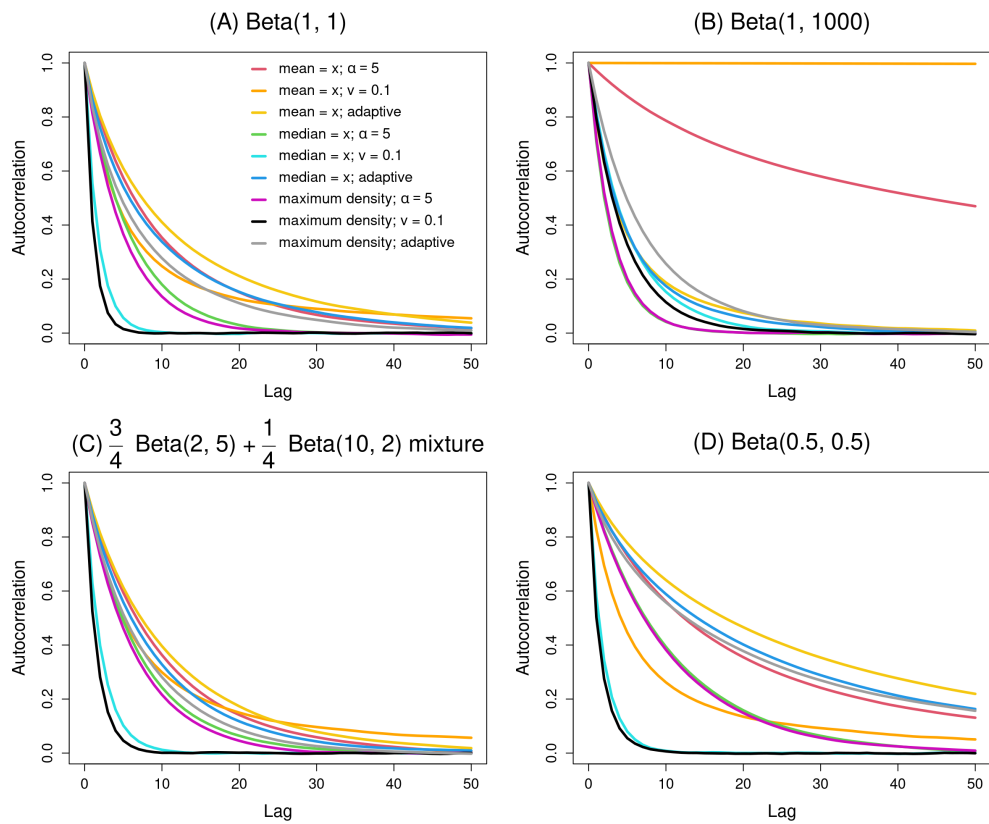
S10

Figure S3: Autocorrelation functions for the mean, median, and maximum density methods of specifying a proposal distribution in the MH example in Section 3.1. Compare with Figure 5.
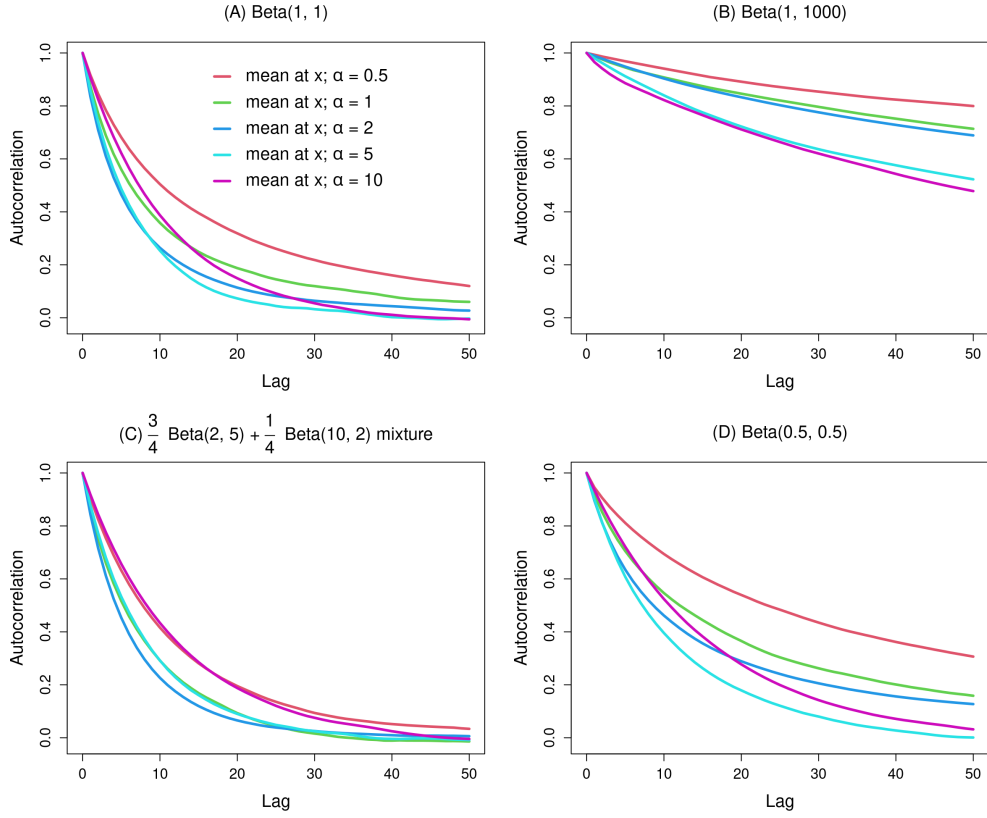
Figure S4: Tuning the concentration parameter of the mean method. Autocorrelation functions for Metropolis–Hasting samplers using proposal distribution II (mean equal to the current state x and fixed concentration parameter α) as in Section 3.1.
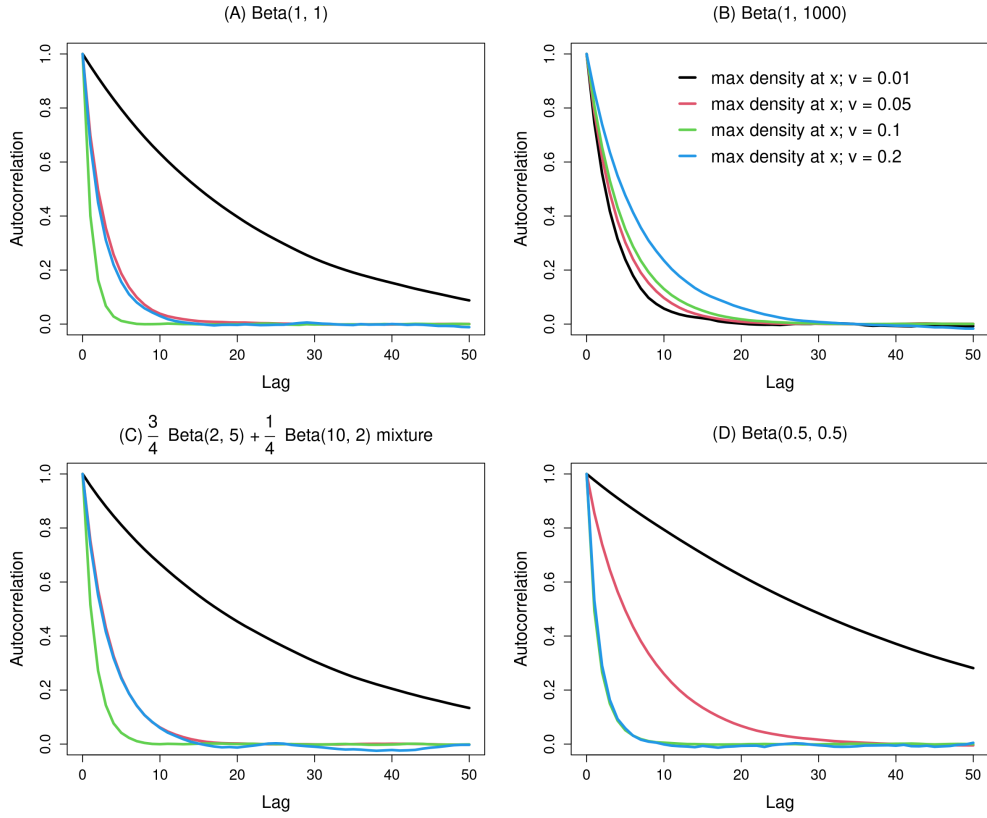
Figure S5: Tuning the variance parameter of the maximum density method. Autocorrelation functions for Metropolis–Hasting samplers using proposal distribution I (maximum density method with target location equal to the current state x and fixed variance v) as in Section 3.1.

(C) $\frac{3}{4}$ Beta(2, 5) + $\frac{1}{4}$ Beta(10, 2)   (D) Beta(0.5, 0.5)

(I) max density at x; v = 0.1   (I) max density at x; v = 0.1

(II) mean = x; α = 5   (II) mean = x; α = 5

(III) mean = x; v = 0.1   (III) mean = x; v = 0.1

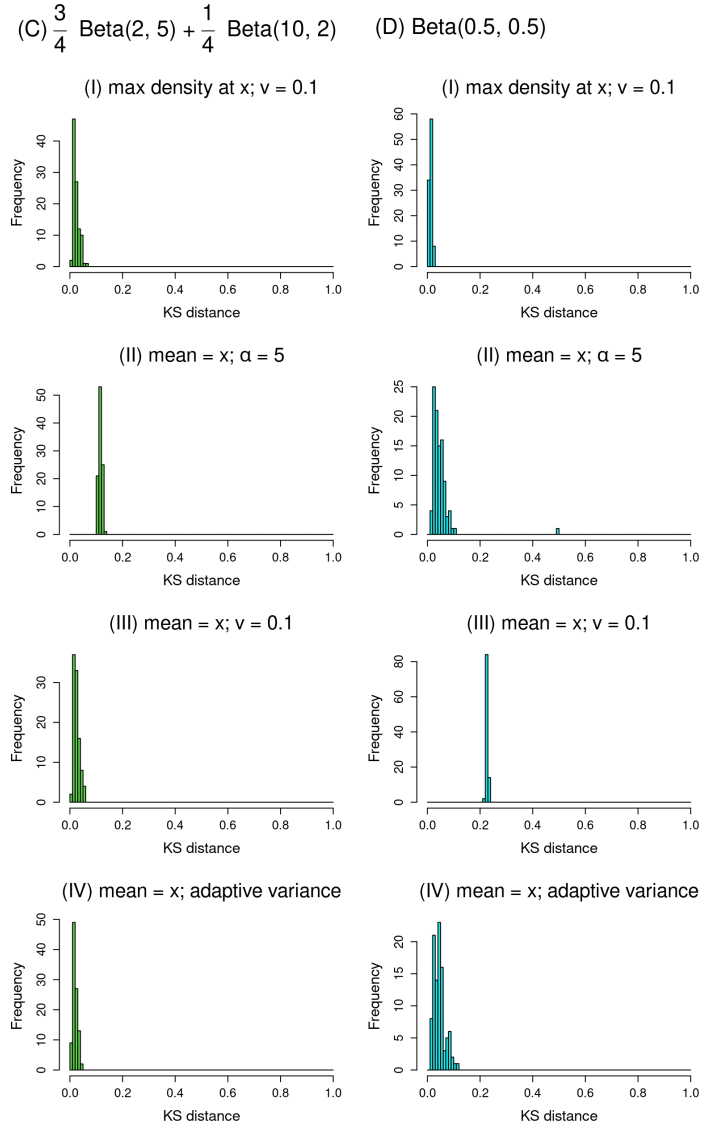(IV) mean = x; adaptive variance   (IV) mean = x; adaptive variance

Figure S6: Histograms of the Kolmogorov–Smirnov distance between the target distribution and the MCMC approximation to it. The green histograms in the left column are for the $0.75\mathrm{Beta}(2,5) + 0.25\mathrm{Beta}(10,2)$ target distribution (case C); the cyan histograms in the right column are for the $\mathrm{Beta}(0.5, 0.5)$ target distribution (case D). As in Figure 6, samplers using maximum density proposals (method I) or the mean method with adaptive variance (method IV) converge reliably to the target distribution, while Methods II and III are less reliable depending on the specific target distribution.

S14

S6 Taylor approximation to cosine error

The cosine error between points x and u in the probability simplex Δ_K is

$$\text{CosErr}(x, u) = 1 - \frac{u^\mathsf{T} x}{\|u\|\|x\|} = 1 - \frac{\sum_i u_i x_i}{\sqrt{\sum_i u_i^2 \sum_i x_i^2}}. \tag{23}$$

To make this expression more tractable when taking its expectation over x, we employ a Taylor approximation to the cosine error as a function of x. To this end, fix $u \in \Delta_K$ and define $f(x) = \text{CosErr}(x, u)$. Since f is uniquely minimized at $x = u$ by the Cauchy–Schwarz inequality, we use the second-order Taylor approximation. To derive this, we compute the first and second derivatives of f, which after simplifying are equal to

$$\frac{\partial f}{\partial x_i} = -\frac{u_i}{\|u\|\|x\|} + \frac{(u^\mathsf{T} x) x_i}{\|u\|\|x\|^3} \tag{24}$$

$$\frac{\partial^2 f}{\partial x_i \, \partial x_j} = \frac{u_i x_j + u_j x_i + (u^\mathsf{T} x)\mathbb{1}(i = j)}{\|u\|\|x\|^3} - \frac{3(u^\mathsf{T} x) x_i x_j}{\|u\|\|x\|^5}. \tag{25}$$

Let $H = (H_{ij}) \in \mathbb{R}^{K \times K}$ denote the Hessian matrix at $x = u$, that is,

$$H_{ij} = \frac{\partial^2 f}{\partial x_i \, \partial x_j}(u) = \frac{\mathbb{1}(i = j)}{\|u\|^2} - \frac{u_i u_j}{\|u\|^4}$$

after plugging $x = u$ into Equation 25 and simplifying. Note that H is symmetric and positive semi-definite since f is minimized at $x = u$. Also, $f(u) = 0$ and $\partial f / \partial x_i = 0$ at $x = u$ by Equations 23 and 24. The second-order Taylor approximation around u is therefore

$$
\begin{aligned}
f(x) &\approx f(u) + \frac{1}{2}(x - u)^\mathsf{T} H (x - u) \\
&= \frac{1}{2} \sum_{i=1}^K \sum_{j=1}^K (x_i - u_i) H_{ij} (x_j - u_j) \\
&= \frac{1}{2} \sum_{i=1}^K (x_i - u_i)^2 \frac{1}{\|u\|^2} - \frac{1}{2} \sum_{i=1}^K \sum_{j=1}^K (x_i - u_i)(x_j - u_j) \frac{u_i u_j}{\|u\|^4}.
\end{aligned}
\tag{26}
$$

Now, suppose $X = (X_1, \ldots, X_K) \sim \text{Dirichlet}(\alpha u_1, \ldots, \alpha u_K)$ where $\alpha > 0$. Then $u = \mathbb{E}(X)$ is the mean of X, so taking the expectation of Equation 26 over X yields

$$\mathbb{E}\big(f(X)\big) \approx \frac{1}{2\|u\|^2} \sum_{i=1}^K \text{Var}(X_i) - \frac{1}{2\|u\|^4} \sum_{i=1}^K \sum_{j=1}^K u_i u_j \text{Cov}(X_i, X_j). \tag{27}$$

Plugging in the formulas for the variances and covariances of the entries of a Dirichlet distributed random vector and simplifying, Equation 27 becomes

$$\mathbb{E}\big(f(X)\big) \approx \frac{1}{2(1 + \alpha)\|u\|^2} \left(1 - \frac{\sum_i u_i^3}{\sum_i u_i^2}\right). \tag{28}$$

S15

Finally, letting $a_i = \alpha u_i$ for $i = 1, \ldots, K$, Equation 28 can be written as

$$
\begin{aligned}
\mathbb{E}(\mathrm{CosErr}(X, \mathbb{E}(X))) = \mathbb{E}(f(X)) &\approx \frac{1}{2(1 + \alpha)\|a/\alpha\|^2}\left(1 - \frac{\sum_i (a_i/\alpha)^3}{\sum_i (a_i/\alpha)^2}\right) \\
&= \frac{\sum_i a_i}{2(1 + \sum_i a_i)(\sum_i a_i^2)}\left(\sum_i a_i - \frac{\sum_i a_i^3}{\sum_i a_i^2}\right).
\end{aligned}
\tag{29}
$$

References

Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press. S4

Casella, G. and Berger, R. (2024). *Statistical Inference*. CRC Press. S3

Ivády, P. (2016). "Corrigendum to: "On a Beta function inequality"." *Journal of Mathematical Inequalities*, 10(2): 597–601. S1

Jacod, J. and Protter, P. (2012). *Probability Essentials*. Springer Science & Business Media. S3

Miller, J. W. and Harrison, M. T. (2014). "Inconsistency of Pitman–Yor process mixtures for the number of components." *The Journal of Machine Learning Research*, 15(1): 3333–3370. S5

Zhao, T. and Wang, M. (2023). "A lower bound for the beta function." *arXiv preprint arXiv:2305.02754*. S1