Statistical Analysis and Data Mining: An ASA Data Science Journal

WILEY

RESEARCH ARTICLE OPEN ACCESS

Mixture-Based Estimation of Multivariate Data Hypervolume

Luca Scrucca

Department of Statistical Sciences, University of Bologna, Bologna, Italy

Correspondence: Luca Scrucca (luca.scrucca@unibo.it)

Received: 5 January 2026 | **Revised:** 16 April 2026 | **Accepted:** 20 May 2026

Keywords: anomaly detection | ecological niche estimation | Gaussian mixtures | GMM-hull | hypervolume estimation | importance sampling | Latin hypercube sampling | multivariate density level sets | noise mixture component

ABSTRACT

Estimating the hypervolume occupied by multivariate data is a fundamental problem in statistics and data science, with applications ranging from ecology and machine learning to multi-objective optimization and Bayesian inference. Traditional approaches rely on geometric approximations, kernel density estimation, or convex-hull constructions, which often suffer from restrictive assumptions or do not scale well in higher dimensions. We introduce a novel methodology for hypervolume estimation based on finite Gaussian mixture models. The proposed approach defines the hypervolume as a high-probability region of the fitted mixture density and estimates its volume using efficient Monte Carlo techniques, such as Latin hypercube sampling and importance sampling. An automatic, data-driven procedure selects the density threshold that determines the region over which the hypervolume is computed. Across simulations, the proposed mixture-based estimator proves broadly applicable and achieves accuracy, flexibility, and computational efficiency equal to or superior to those of existing methods. Applications to anomaly detection and ecological niche estimation illustrate the method's practical utility and interpretability in complex multivariate settings.

1 | Introduction

The hypervolume of a multivariate dataset refers to the volume of a multidimensional space that is enclosed by a set of data points. By providing a geometric quantification of the space occupied by a multivariate dataset, the hypervolume is a useful concept in statistics and data science, enabling assessments of diversity, spread, and coverage in high-dimensional feature spaces.

Estimation of the data hypervolume plays an important role in several data analysis applications, including ecology, machine learning, multi-objective optimization, and Bayesian model selection. In ecological studies, hypervolumes provide a multidimensional representation of ecological niches or community traits. Rather than focusing on a single factor, hypervolumes simultaneously consider multiple environmental and biological variables, enabling researchers to capture the complexity of ecosystems beyond one-dimensional or simple summary metrics. This approach is crucial for understanding ecological strategies, biodiversity, and ecosystem stability [1]. Anomaly detection in machine learning algorithms is important for identifying patterns in data that deviate from regular conditions. Outlying or noisy data points can be detected through mixture modeling by assuming a uniform distribution for the noise component, which in turn requires estimating the hypervolume of a multivariate dataset [2]. The hypervolume is also widely used as a performance metric for evaluating stochastic multi-objective optimizers and for selection processes in evolutionary multi-objective optimization algorithms [3]. More recently, Metodiev et al. [4] introduced a method that employs the hypervolume when computing the truncated harmonic mean estimator (THAMES) of the inverse marginal likelihood, a quantity required for Bayesian hypothesis

testing via Bayes factors. These diverse applications highlight the methodological importance of hypervolume estimation as an interpretable, geometry-based approach to multivariate analysis.

Several methods have been developed for hypervolume estimation. Traditional approaches rely on geometric approximations, such as hyperrectangles aligned with the original axes or along the principal components, and hyperellipsoids derived from elliptical distributions, such as the multivariate Gaussian. A more straightforward approach to estimating the hypervolume is to compute the volume of the convex hull, a concept from computational geometry defined as the smallest convex set enclosing the data points [5]. Determining the d-dimensional convex hull is a computationally challenging problem, and the Quickhull algorithm [6] represents the standard method for its calculation. However, its practical use is limited to low-dimensional settings (typically 2D, 3D, and up to about 4D or 5D), as the algorithm's complexity increases exponentially with dimensionality, making computation prohibitively expensive.

Methods based on kernel density estimation (KDE) have become popular for constructing nonparametric estimates of probability density functions, offering flexible shape adaptation and requiring fewer distributional assumptions [7, 8]. Parametric approaches have also been proposed, typically assuming a multivariate Gaussian distribution [9], with subsequent extensions incorporating random effects [10]. Monte Carlo sampling techniques provide an additional strategy for approximating hypervolumes, particularly in high-dimensional settings where direct computation is infeasible. In the domain of multi-objective evolutionary optimization, hypervolume-based metrics are central for assessing solution quality, and algorithms such as HypE offer efficient approximation procedures that balance accuracy with computational cost [11].

Despite the variety of approaches discussed above, hypervolume estimation remains challenging, especially as dimensionality increases. Geometric estimators based on range-boxes, PCA projections, or hyperellipsoidal approximations impose rigid assumptions on the shape of the data support that rarely accommodate the complex patterns typical of real datasets. Kernel density-based estimators offer greater flexibility but become computationally expensive for large samples and are highly sensitive to bandwidth selection. Monte Carlo methods provide an alternative but often require prohibitively large sample sizes to deliver accurate volume estimates in high dimensions. Although a range of algorithms and approximations have been proposed to address such limitations, balancing accuracy and computational efficiency remains a central concern in practical applications.

In this paper, we propose a novel methodology based on finite Gaussian mixtures for estimating the hypervolume of a multivariate dataset. Our proposal aims to overcome the main limitations of existing techniques and to prove robust performance across diverse multivariate datasets. Through comparative analyses and applied examples, we show that the proposed method improves accuracy, computational efficiency, and flexibility in handling complex multivariate data.

2 | Methodology

2.1 | The Hypervolume of a Multivariate Dataset

For a multivariate distribution defined in the d-dimensional space, $X \in \mathcal{X} \subset \mathbb{R}^d$, the hypervolume corresponds to the geometric size of the region where the distribution is supported. In the bounded case, this can be expressed as the multiple integral

$$V(X) = \int_{\mathcal{X}} \mathrm{d}x_1 \, \mathrm{d}x_2 \ldots \mathrm{d}x_d.$$

When the support $\mathcal{X}$ of the distribution is unbounded, the hypervolume is infinite, and one usually considers the volume of probability regions containing a prescribed probability mass.

Given a dataset $\{\boldsymbol{x}_i\}_{i=1}^n \in \mathcal{X}$ of n observations on d variables, the hypervolume of the dataset can be regarded as the empirical analogue of the distributional hypervolume. Define the smallest axis-aligned hyperrectangle enclosing the data as

$$\mathcal{H} = \{\boldsymbol{x} \in \mathcal{X} : \ell_j \leq x_j \leq u_j \quad \text{for all } j = 1, \ldots, d\},$$

where ℓ_j and u_j denote, respectively, the observed lower and upper bounds for each dimension, that is, $\ell_j = \min\{x_{ij}\}_{i=1}^n$ and $u_j = \max\{x_{ij}\}_{i=1}^n$ for $j = 1, \ldots, d$. A simple estimate of the hypervolume can then be obtained by considering the volume of the d-dimensional hyperrectangle enclosing the sample, that is,

$$V_{\mathcal{H}} = \int_{\mathcal{H}} \mathrm{d}\boldsymbol{x} = \prod_{j=1}^{d} (u_j - \ell_j).$$

The underlying assumption behind this estimate of the hypervolume is that the data are drawn from a multivariate uniform distribution. In practice, however, the observed data rarely occupy the entire d-dimensional hyperrectangle. Instead, they typically fill only a small portion of the space, a fraction that decreases rapidly as dimensionality increases, giving rise to the so-called "curse of dimensionality." For this reason, the volume of the convex hull of the observed data points would, in principle, provide a more appropriate estimate. Nevertheless, as discussed in the introduction, computing the convex hull becomes quickly infeasible as dimensionality grows. Hence, a more general and scalable solution to this estimation problem is required.

For reasons of numerical stability, in particular to mitigate underflow and overflow issues in high-dimensional settings, computations are typically performed on the log-scale. For instance, we can express the log-volume of the d-dimensional hyperrectangle as

$$\log V_{\mathcal{H}} = \sum_{j=1}^{d} \log(u_j - \ell_j).$$

Accordingly, in the remainder of this paper, we usually compute the log-volume, but due to the monotonicity of the log-transformation, the volume can be readily recovered by exponentiation when required. It is interesting to note that for the multivariate uniform distribution the log-volume is equal to the differential entropy, a measure of the amount of uncertainty

or randomness present in the dataset [12]. Hence, large values of log-volume indicate greater uncertainty and, consequently, greater dispersion within the dataset.

The main idea at the basis of our proposal for estimating the volume of a multivariate dataset is to adjust the volume of the hyperrectangle enclosing the observed data by a scalar quantity, say $\kappa \in (0, 1]$, such that

$$V = V_{\mathcal{H}}\kappa. \tag{1}$$

When working in the logarithmic scale, for the reasons mentioned above, this is equivalent to compute

$$\log(V) = \log\big(V_{\mathcal{H}}\big) + \log(\kappa).$$

As final remark, we note that in this paper the target quantity is the empirical data hypervolume, defined relative to the region containing the observed data, rather than the unrestricted level set of the distribution over $\mathbb{R}^d$. This choice is deliberate and it prevents extrapolating from regions unsupported by the observed data.

2.2 | Estimation of Hypervolume Based on Gaussian Mixtures

Finite mixture models are a class of probabilistic models that represent a distribution as a convex combination of simpler component distributions [13, 14]. Gaussian mixture models (GMMs) are obtained by assuming Gaussian densities for the mixture components, so a G-component GMM can be written as

$$f(\boldsymbol{x}) = \sum_{k=1}^{G} \pi_k \phi\big(\boldsymbol{x}; \boldsymbol{\mu}_k, \Sigma_k\big), \tag{2}$$

where π_k are the mixing coefficients (under the constraints $\pi_k > 0$ and $\sum_{k=1}^{G}\pi_k = 1$), $\phi()$ is the multivariate Gaussian density function with parameters $\boldsymbol{\mu}_k$, the d-dimensional vector of means, and Σ_k, the $(d \times d)$ covariance matrix, for the k-th component $(k = 1, \ldots, G)$. Parsimonious covariances decomposition can be adopted by imposing constraints on the geometric characteristics, such as volume, shape, and orientation, of corresponding ellipsoids as proposed by Banfield and Raftery [15], Celeux and Govaert [16]. See also Scrucca et al. [17, Section 2.2.1]. Moreover, to prevent degeneracies in the likelihood, a regularizing conjugate prior can be adopted, as described by Fraley and Raftery [18].

Suppose the data distribution can be approximated by a GMM, with the number of components and the covariance parameterization selected using the Bayesian Information Criterion (BIC; [19, 20]). Parameter estimation is typically carried out via the Expectation–Maximization (EM) algorithm [21], yielding maximum a posteriori (MAP) estimates when the prior regularization is included. For sufficiently large samples, these estimates are essentially equivalent to maximum likelihood estimates (MLEs).

Note that the choice of mixture model specification affects the estimated density level set and hence the hypervolume estimate. As mentioned, in this work BIC is used as an automatic default because it offers a stable, data-driven compromise between flexibility and parsimony, yielding a sufficiently accurate approximation of the density level set. In some cases, a mildly overfitted mixture may better capture local boundary geometry, whereas in other cases it may introduce spurious components or artificial low-density connections. For this reason, the BIC-selected model should be regarded as a pragmatic default, and sensitivity checks over nearby mixture specifications may be informative.

Having fitted a GMM to the observed data $\{\boldsymbol{x}_i\}_{i=1}^{n}$, a GMM-based estimate of the data hypervolume can be obtained as

$$V_{\text{GMM}} = \int_{\mathcal{H}} \mathbb{1}(\hat{f}(\boldsymbol{x}) \geq h)\mathrm{d}\boldsymbol{x}, \tag{3}$$

where $\mathbb{1}()$ is the indicator function taking value 1 if the condition is true, and 0 otherwise, and $\hat{f}(\boldsymbol{x})$ is the density estimate from GMM in (2), that is,

$$\hat{f}(\boldsymbol{x}) = \sum_{k=1}^{G} \hat{\pi}_k \phi\Big(\boldsymbol{x}; \hat{\boldsymbol{\mu}}_k, \hat{\Sigma}_k\Big), \tag{4}$$

with estimated parameters $\left\{\hat{\pi}_k, \hat{\boldsymbol{\mu}}_k, \hat{\Sigma}_k\right\}_{k=1}^{G}$, and h is a density threshold value that needs be set, as discussed in Section 2.2.3.

In practice, computing integrals over indicator functions often relies on numerical methods, as closed-form solutions are typically unavailable. One effective approach is to rewrite the above integral as follows:

$$\begin{aligned} V_{\text{GMM}} &= V_{\mathcal{H}} \int_{\mathcal{H}} \mathbb{1}(\hat{f}(\boldsymbol{x}) \geq h) V_{\mathcal{H}}^{-1}\mathrm{d}\boldsymbol{x} \\ &= V_{\mathcal{H}} \Pr_{X \sim U(\mathcal{H})}(f(X) \geq h) = V_{\mathcal{H}}\kappa \end{aligned}$$

where $\kappa \in (0, 1]$ expresses the probability that the GMM density is not less than the threshold value h when X is uniformly distributed over the hyperrectangle $\mathcal{H}$. This estimator matches the form in (1), where the volume of the hyperrectangle is scaled by the constant κ. In the following, we discuss two Monte Carlo sampling proposals for efficiently estimating κ.

2.2.1 | Latin Hypercube Sampling Approach

A straightforward way to estimate κ is by Monte Carlo (MC) simulation. Consider a set of simulated data points $(\tilde{\boldsymbol{x}}_1, \ldots, \tilde{\boldsymbol{x}}_S)$ drawn from $U(\mathcal{H})$, then the MC estimate is given by

$$\hat{\kappa} = \frac{1}{S}\sum_{s=1}^{S} \mathbb{1}\big(f(\tilde{\boldsymbol{x}}_s) \geq h\big)$$

where $\left\{\hat{f}(\tilde{\boldsymbol{x}}_s)\right\}_{s=1}^{S}$ are the mixture density values evaluated at the S simulated data points. By the Law of Large Numbers, this estimate converges almost surely to the true value of κ as the number of draws S tends to infinity.

Sampling uniformly from $\mathcal{H}$ can be obtained by simple Monte Carlo (SMC) sampling, that is, by drawing independent uniform samples within the hyperrectangle bounds. A better space-filling sampling scheme can be obtained by using Latin hypercube sampling (LHS), a more efficient method that generates pseudo-random samples via stratified random sampling

[22–24]. For the j-th data dimension ($j = 1, \ldots, d$), the LHS algorithm generates S draws using the following procedure:

i. Generate a random permutation $\tilde{x}_{ij}$ of the integers $1, \ldots, S$;

ii. Generate S independent draws from uniform distribution, $v_i \sim U(0,1)$, for $i = 1, \ldots, S$;

iii. Construct stratified draws as follows:

$$\tilde{x}_{ij} \leftarrow \frac{\tilde{x}_{ij} - 1 + v_i}{S}, \quad i = 1, \ldots, S$$

iv. Rescale to the range of j-th dimension, $[\ell_j, u_j]$, using

$$\tilde{x}_{ij} \leftarrow \ell_j + (u_j - \ell_j)\tilde{x}_{ij}$$

By dividing the range of each variable into equal intervals and ensuring that each interval is sampled exactly once, LHS provides better coverage of the input space compared to SMC, since the latter may leave some regions under- or over-sampled due to its purely random nature. As a result, LHS acts as a variance reduction technique that can significantly reduce the number of simulation runs needed to achieve comparable accuracy relative to standard SMC methods.

A Laplace smoothing correction could also be applied when computing the estimate, yielding

$$\widehat{\kappa} = \frac{1 + \sum_{s=1}^{S} \mathbb{1}\left(\widehat{f}(\tilde{\mathbf{x}}_i) \geq h\right)}{S + 2}.$$

This is known to be equivalent to assuming a Beta$(1,1)$ prior on the proportion, which has the effect of shrinking the estimate toward 0.5, but for large S the effect is negligible, while it guarantees that $\widehat{\kappa} \in (0,1)$.

Finally, the hypervolume of the data can be estimated as follows:

$$\widehat{V}_{\text{GMM}}^{\text{LHS}} = V_{\mathcal{H}}\widehat{\kappa}$$

2.2.2 | Importance Sampling Approach

This MC approach based on LHS works well in low dimensions, but as d grows most of the hyperrectangle space $\mathcal{H}$ is empty of data points, so $\widehat{\kappa}$ approaches zero and the resulting estimates become unstable.

Importance sampling [25] is an alternative variance reduction technique that can be used to improve the efficiency of Monte Carlo integration. LHS spreads samples evenly across the input space but does not focus on important regions explicitly. On the contrary, importance sampling draws samples from regions of the input space where the integrand is large or contributes most to the integral, hence improving estimator efficiency.

Consider the MC estimator in (3) by rewriting it as an expectation:

$$\begin{aligned} V_{\text{GMM}}^{\text{IS}} &= \int_{\mathcal{H}} \mathbb{1}(f(\mathbf{x}) \geq h)\mathrm{d}\mathbf{x} = \int_{\mathcal{H}} \frac{\mathbb{1}(f(\mathbf{x}) \geq h)}{q(\mathbf{x})} q(\mathbf{x})\mathrm{d}\mathbf{x} \\ &= \mathbb{E}_q\left[\frac{\mathbb{1}(f(X) \geq h)}{q(X)}\right] \end{aligned}$$

for any proposal density $q()$ with support covering $\mathcal{H}$. Assuming the uniform density over $\mathcal{H}$, $q(\boldsymbol{x}) = 1/V_{\mathcal{H}}$ if $\boldsymbol{x} \in \mathcal{H}$ and 0 otherwise, then the previous LHS sampling case is obtained. However, this formulation allows to introduce an importance sampling approximation of the volume by computing

$$V_{\text{GMM}}^{\text{IS}} = \frac{1}{S}\sum_{s=1}^{S} \frac{\mathbb{1}\left(f(\tilde{\boldsymbol{x}}_s) \geq h\right)}{q(\tilde{\boldsymbol{x}}_s)}$$

where $\{\tilde{\boldsymbol{x}}_s\}_{s=1}^{S}$ are drawn from the distribution with density $q()$. This estimator is unbiased for any proposal density $q()$, while the variance depends on how well the support of $q()$ overlaps with the region $\mathcal{H}$.

A natural choice for the proposal density $q()$ is to use the density estimated by GMM, which is already concentrated where the data lie. Then, an importance sampling estimate of the volume is computed as

$$\widehat{V}_{\text{GMM}}^{\text{IS}} = \frac{1}{S}\sum_{s=1}^{S} \frac{\mathbb{1}\left(f(\tilde{\boldsymbol{x}}_s) \geq h \wedge \tilde{\boldsymbol{x}}_s \in \mathcal{H}\right)}{f(\tilde{\boldsymbol{x}}_s)}$$

where $\{\tilde{\boldsymbol{x}}_s\}_{s=1}^{S}$ are simulated from the estimated GMM and truncated to lie in the support of the data $\mathcal{H}$. Note that it is not necessary to explicitly calculate κ, but, if required, it can be easily derived by observing that, by definition, $\widehat{\kappa} = \widehat{V}_{\text{GMM}}^{\text{IS}}/V_{\mathcal{H}}$.

2.2.3 | Selection of Threshold Level

The problem of selecting the threshold value h can be recast as finding the level set of the highest density region (HDR) for a specific target probability [26]. Given $\alpha \in (0,1]$, the α-HDR is the set

$$R_\alpha = \left\{\boldsymbol{x} \in \mathbb{R}^d : f(\boldsymbol{x}) \geq h_\alpha\right\},$$

where the threshold h_α is defined implicitly as

$$\int_{R_\alpha} f(\boldsymbol{x})\mathrm{d}\boldsymbol{x} = \Pr_{X \sim f}\left(f(X) \geq h_\alpha\right) = \alpha,$$

which amounts to choose the smallest density threshold h_α such that $\Pr_{X\sim f}\left(f(X) \geq h_\alpha\right) \geq \alpha$. Thus, R_α encloses the most probable α fraction of the probability mass and, among all sets with that probability content, has minimal volume.

In practice, given a sample $\{\boldsymbol{x}_i\}_{i=1}^{n}$ assumed to be drawn from the mixture density in (2), the density threshold h_α is estimated as the $(1-\alpha)$-quantile of the empirical distribution. For $\alpha = 1$, the 100% HDR corresponds to setting $h_\alpha = \min\left\{\widehat{f}(\boldsymbol{x}_i)\right\}_{i=1}^{n}$, the minimum estimated density over the observed data points. However, this can be extremely sensitive to outliers and data sparsity in high-dimensional cases. A more robust alternative is to select a value of $\alpha \in [0.9, 0.999]$. A default density threshold is determined automatically by fitting a two-segment piecewise linear regression model (i.e., a model with a single change point) to a grid of (h_α, α) values computed for HDR levels in the range [0.9,1]. The two-segment piecewise linear fit should be interpreted as a heuristic device for identifying a transition in the (h_α, α) curve, rather than as evidence of a true

structural break. In smoothly decaying distributions, the selected change-point provides a convenient default summarizing the onset of a tail-dominated regime. Alternative choices of α may be considered when substantive considerations suggest a different threshold.

Finally, we can define the *GMM hull* as the density level set corresponding to the region containing probability mass α under the fitted GMM. The special case $\alpha = 1$ yields the largest level set that encloses all observations.

2.2.4 | An Illustrative Example

In this section we provide a detailed example of the proposed approach for hypervolume estimation. Consider a dataset drawn from a uniform distribution inside a d-dimensional hyperellipsoid. In general, a hyperellipsoid in $\mathbb{R}^d$ centered at $\boldsymbol{\mu}$ with positive definite shape matrix $\boldsymbol{A}$ can be written as

$$E = \left\{\mathbf{x} \in \mathbb{R}^d \,:\, (\mathbf{x}-\boldsymbol{\mu})^\top \mathbf{A}^{-1}(\mathbf{x}-\boldsymbol{\mu}) \le 1\right\}.$$

The log-volume of this hyperellipsoid is given by

$$\log V = \frac{d}{2}\log(\pi) - \log\Gamma(d/2+1) + \frac{1}{2}\log(\det(\boldsymbol{A})).$$

Figure 1a shows the hyperellipsoid in 2D centered at $\boldsymbol{\mu} = [0,0]^\top$ and with $\boldsymbol{A} = \begin{bmatrix} 1.0 & 0.8 \\ 0.8 & 1.0 \end{bmatrix}$, for which the log-volume is easily computed as $\log V \approx 0.6339$. Figure 1b displays a bivariate sample generated from this hyperellipsoid, together with the associated data hyperrectangle given by the smallest boxed region with sides parallel to the coordinate axes containing all data points. The corresponding $\log V_H \approx 1.3655$ is larger than the true log-volume, since a large portion of the hyperrectangle contains no data points. A refinement is obtained by computing the volume of the smallest hyperrectangle aligned with the principal axes, which yields $\log V_{\text{PC}} \approx 0.8485$. However, empty regions remain, leading again to an overestimate. A better alternative is to compute the volume enclosed by the convex hull of the data, which in this bidimensional case is feasible and yields $\log V_{\text{CHULL}} \approx 0.6075$, a slight underestimate of the true log-volume.

Figure 1c–f illustrate our proposal. Figure 1c reports the graph of the HDR density h_α versus the HDR level $\alpha \in [0.9,1]$ used for the automatic selection of threshold value: the two-segment piecewise linear fit (gray thick line) identifies the default h_α value corresponding to the change-point of the piecewise model (red dashed lines), along with the value defining the 100% HDR region (orange dotted lines). The corresponding GMM hulls for these thresholds are shown in Figure 1d.

Figure 1e,f show the MC draws obtained using, respectively, the LHS and IS Monte Carlo procedures described in Section 2.2, both employing the automatically selected threshold h_α. In this case, both methods yield a slight underestimate of the true log-volume, but with improved accuracy compared to that of the convex hull procedure. However, this will not necessarily occur in general. Note that using the smallest density value instead would result in larger, less accurate log-volume estimates (see values in parentheses).

2.2.5 | Some Remarks on Monte Carlo Sample Size and Computing Time

To investigate the effect of the MC sample size on estimation accuracy and computational time of the proposed approach, we conducted a simulation study that extends the data-generating mechanism described in the previous section to higher dimensions. Specifically, a correlated hyperellipsoid was generated, as detailed in Section 3.4. We considered sample sizes $n \in \{200, 1000\}$, data dimensions $d = \{2,10,50\}$, and MC sample sizes $S = \left\{10^3, 10^4, 10^5, 10^6\right\}$. The experiment was replicated 100 times for each combination of n, d, and S. In each replication, we recorded the mean absolute percentage error (MAPE; see Equation (5)) and the runtime (in seconds) of the log-hypervolume estimate for both the LHS and IS approaches. Simulations were carried out on a MacBook Pro equipped with an Apple M3 Pro CPU (12 cores, 4.06 GHz) and 36 GB of RAM.

Figure 2 reports the MAPE for increasing MC sample sizes (S) at selected values of the data dimensionality (d) and sample size (n). In the low dimensional case ($d = 2$), the accuracy of LHS and IS is essentially equivalent. As the dimensionally increases, however, the IS approach becomes substantially more accurate and stable than LHS. In particular, for the largest dimensionality investigated, IS clearly outperforms LHS, and its accuracy improves as the MC sample size increases. By contrast, the performance of LHS deteriorates markedly as the dimensionality grows. These findings reflect the inefficiency of uniform sampling in high-dimensional spaces, whereas IS remains comparatively stable because samples are drawn from regions of high density. Overall, selecting an extremely large MC sample size appears unnecessary in low to moderate dimensions, whereas larger values become beneficial when the dimensionality is high relative to the available sample size.

Figure 3 displays the runtime as a function of the MC sample size (S), data dimensionality (d), and number of observations (n). Computation time increases only slightly with S, with a more noticeable effect emerging only in the highest-dimensional setting ($d = 50$). Overall, computation times remain modest across all scenarios considered, typically below a few seconds even for $S = 10^6$. No substantial differences are observed between LHS and IS in terms of runtime across most scenarios, except in the most demanding settings, where IS appears marginally slower. The most noticeable differences arise when moving from the smaller to the larger sample size n, reflecting the increased computational cost of GMM fitting, whereas increasing the MC sample size S has a comparatively minor impact on the overall runtime. These results indicate that the proposed approach scales well with respect to both dimensionality and MC sample size, making it computationally feasible even in moderately high-dimensional settings.

Based on these results and our empirical experience across different datasets, we recommend $S = 10^6$ as a conservative default choice. This value provides a robust trade-off between accuracy

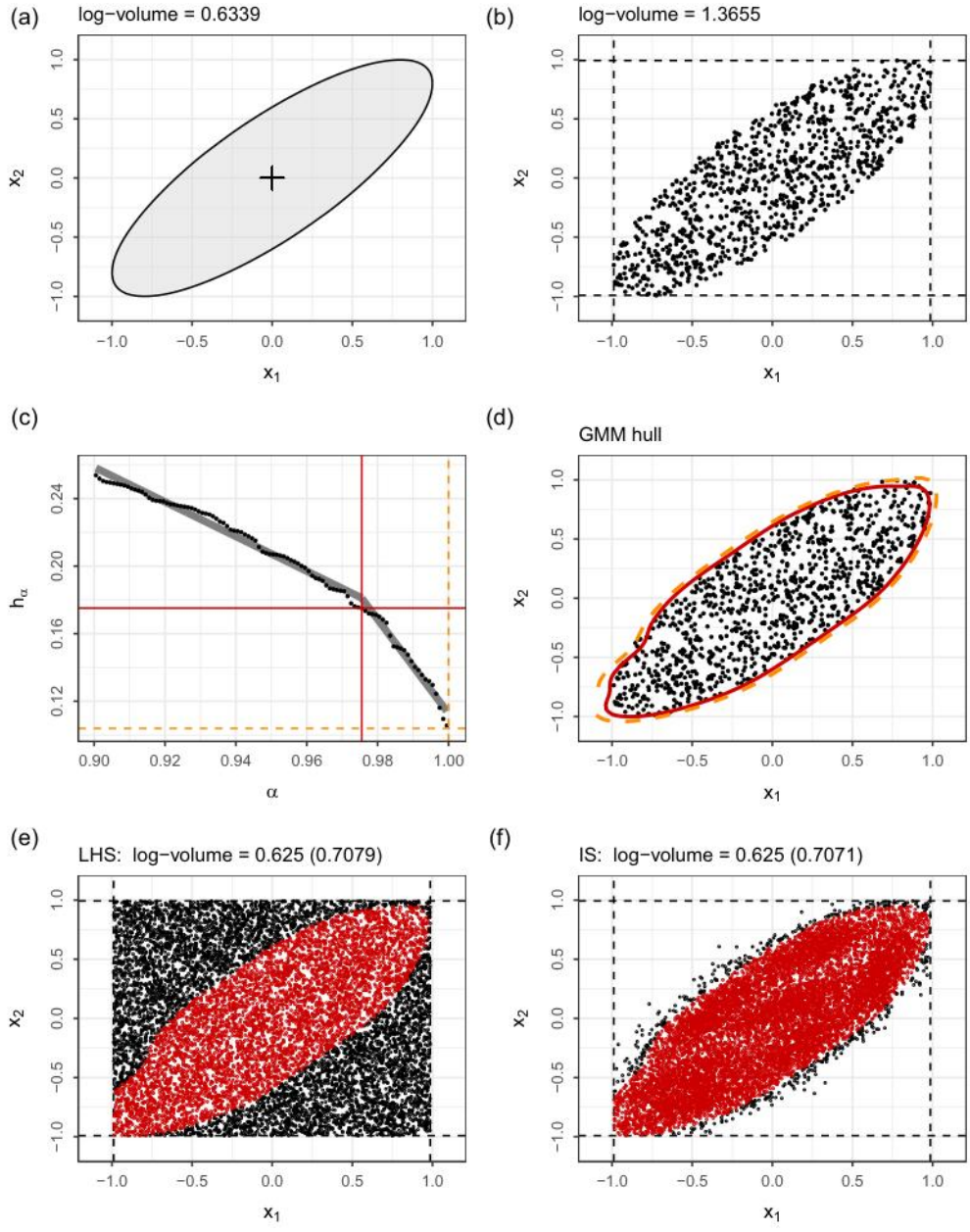

FIGURE 1 | Illustration of the proposed approach to estimating the volume of a 2D hyperellipsoid: (a) hyperellipsoid centered at $\mu = [0, 0]^\top$ with $\boldsymbol{A} = \begin{bmatrix} 1.0 & 0.8 \\ 0.8 & 1.0 \end{bmatrix}$, and $\log V \approx 0.6339$; (b) bivariate sample drawn from the hyperellipsoid with axis-aligned data hyperrectangle and corresponding log-volume ≈ 1.3655; (c) graph of the density threshold h_α as a function of HDR level α, used for automatic threshold selection; the gray line shows the fitted two-segment piecewise linear model; the red vertical and horizontal solid lines identify the selected threshold corresponding to the change-point at $\alpha = 0.9755$; the orange dashed lines correspond to the case $\alpha = 1$; (d) GMM hulls corresponding to the identified threshold levels; (e–f) Monte Carlo draws using the LHS and IS procedures proposed for volume estimation using the automatically selected threshold h_α, with the associated log-volume estimates (values in parentheses refer to the 100% HDR region).

and computational cost and is more than sufficient in most practical scenarios.

3 | Simulation Studies

To assess the empirical performance of the proposed GMM–based estimator of data hypervolume, we conducted an extensive set of simulation experiments. These simulations were designed to evaluate the estimator's accuracy, robustness, and scalability under controlled conditions with known ground truth. By systematically varying data geometry, dimensionality, and sample size, we aim to show the flexibility and efficiency of the proposed approach relative to conventional geometric and nonparametric estimators.

In the following, we present the simulation design, describe the competing methods used for comparison, and summarize the quantitative results obtained across a range of cases using synthetic data.

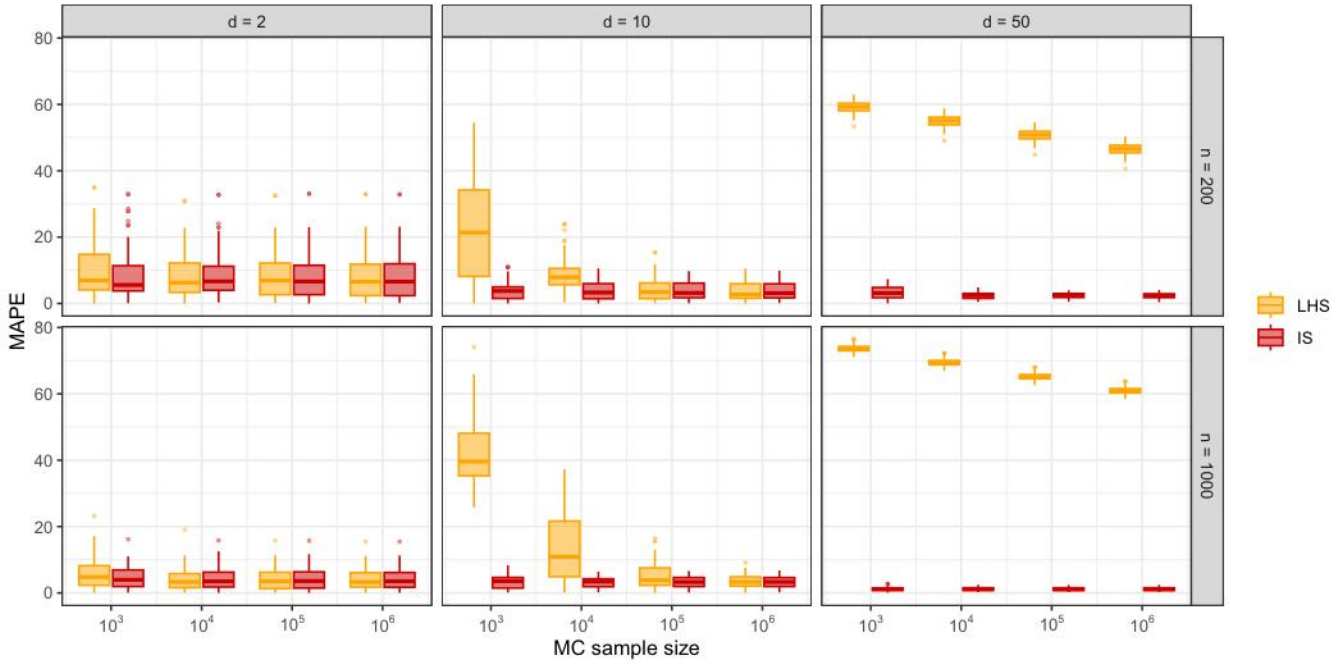

FIGURE 2 | Box-plots of hypervolume estimation accuracy (measured by MAPE) as a function of MC sample size (S), data dimensionality (d), and sample size (n), for the proposed LHS and IS Monte Carlo approaches.

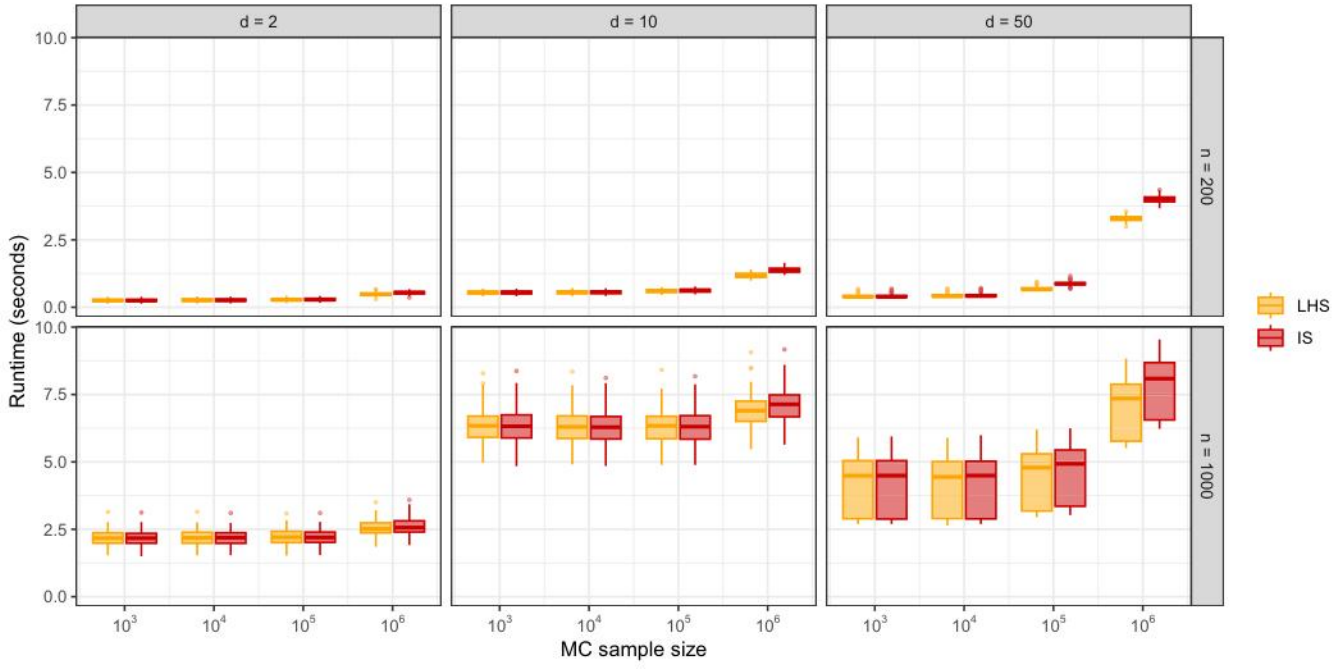

FIGURE 3 | Box-plots of hypervolume estimation runtime (in seconds) as a function of MC sample size (S), data dimensionality (d), and sample size (n), for the proposed LHS and IS Monte Carlo approaches.

3.1 | Experimental Setup

Several synthetic data scenarios were considered, each representing a distinct geometric configuration in $\mathbb{R}^d$: (1) hyperbox, (2) hypersphere, (3) hyperellipsoid, (4) simplex, (5) overlapping hyperspheres and (6) separated hyperspheres. For each configuration, datasets of varying dimensions $d \in \{2,3,5,10,20\}$ and sample sizes $n \in \{100,200,500,1000,2000\}$ were generated. The true hypervolume for each case was computed analytically to provide a ground truth reference.

To assess performance, we compared the proposed GMM-based estimators using Latin hypercube sampling and importance sampling, as described in Section 2.2, both with fixed and automatically selected threshold level, to a set of established estimators. In particular, we considered:

- `HBOXhull`: volume of hyperrectangle enclosing the data, available in the `mclustAddons` R package [27];
- `PCBOXhull`: volume of hyperrectangle aligned with the principal components enclosing the data, available in the `mclustAddons` R package [28];
- `Chull`: convex-hull, available in the `geometry` R package [29];
- `Ehull`: ellipsoidal hull, available in the `cluster` R package [30];
- `TMVNhull`: truncated multivariate normal, available in the `mclustAddons` R package [28];
- `HypBox`: volume estimation via hyperbox kernel density estimate, available in the `hypervolume` R package [31];
- `HypGauss`: volume estimation via Gaussian kernel density estimate, available in the `hypervolume` R package [31];
- `HypSVM`: volume estimation via one-class support vector machine learning model, available in the `hypervolume` R package [31];
- `GMMhull[LHS-1]`: volume estimation via Gaussian mixtures using Latin hypercube sampling with fixed threshold level corresponding to $\alpha = 1$, available in the `mclustAddons` R package [28];
- `GMMhull[LHS]`: volume estimation via Gaussian mixtures using Latin hypercube sampling with automatic threshold level selected as discussed in Section 2.2.3, available in the `mclustAddons` R package [28];
- `GMMhull[IS-1]`: volume estimation via Gaussian mixtures using importance sampling with fixed threshold level corresponding to $\alpha = 1$, available in the `mclustAddons` R package [28];
- `GMMhull[IS]`: volume estimation via Gaussian mixtures using importance sampling with automatic threshold level selected as discussed in Section 2.2.3, available in the `mclustAddons` R package [28].

For each experimental setup, the estimation accuracy was quantified using the mean absolute percentage error (MAPE) computed on the log-volume scale over $R = 100$ replications, that is,

$$\text{MAPE}(\log \widehat{V}) = \frac{1}{R}\sum_{r=1}^{R}\left|\frac{\log\left(\widehat{V}_r\right) - \log(V)}{\log(V)}\right| \times 100, \qquad (5)$$

where $\log(V)$ is the true log-hypervolume for the specific setup, and $\log\left(\widehat{V}_r\right)$ the corresponding estimate at the r-th replication.

In the remainder of this section, we examine several scenarios characterized by different underlying data shapes. Visual examples of the corresponding two-dimensional cases are provided in Appendix A (see Figures A1–A6) to offer qualitative insight into the data-generating mechanisms.

3.2 | Hyperbox Scenario

This scenario is obtained by generating random data points from a uniform distribution within an axis-aligned d-dimensional box with vertices $[-1, 1]$ along each coordinate direction. The true log-volume in this case is given by $\log V = d \log(2)$.

As shown in Figure 4, both the convex hull estimator (`Chull`) and the kernel-based estimators (`HypBox`, `HypGauss`, `HypSVM`) cannot be computed for $d \geq 10$. This limitation appears consistently across all simulation scenarios and represents a crucial constraint of these methods in high-dimensional settings. When they are computable, `Chull` and `HypSVM` perform reasonably well in low dimensions, but their accuracy depends heavily on sufficiently large sample sizes as dimensionality increases. Among the geometric estimators, the axis-aligned hyperrectangle estimator (`HBOXhull`) is, unsurprisingly, the most accurate in

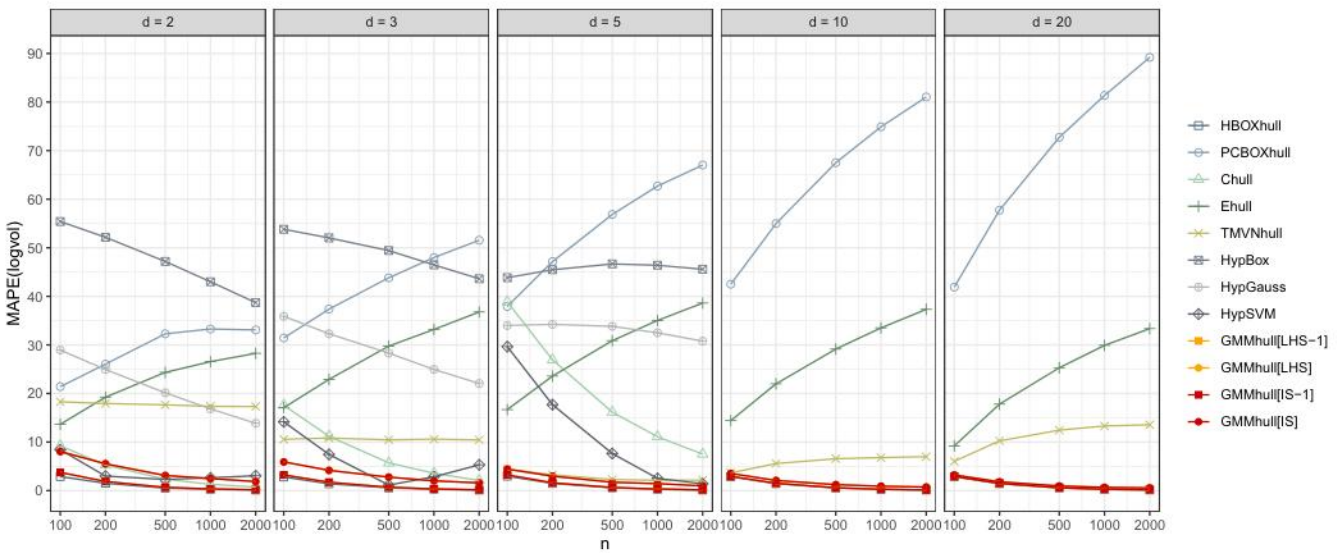

FIGURE 4 | Mean absolute percentage error (MAPE) of log-volume estimates for datasets uniformly distributed within a d-dimensional hyperbox. Results are reported for varying sample sizes (n) and dimensions (d).

this setting, as it matches the true data-generating geometry. In contrast, its principal-component-adjusted counterpart (`PCABOXhull`) is the least accurate and its performance deteriorates markedly with increasing dimension. The GMM-based estimators (`GMMhull[LHS]` and `GMMhull[IS]`) produce nearly unbiased estimates across all tested dimensions, consistently achieving MAPE values below 10%, and performing comparably to the best results obtained by `HBOXhull`. Moreover, GMM-based estimators using either a fixed threshold or the automatically selected threshold appear essentially indistinguishable, with slightly better performance observed for the former. Overall, the GMM-based estimators clearly outperform all competitors except `HBOXhull`, which performs well mainly because its shape corresponds exactly to how the data were generated.

3.3 | Hypersphere Scenario

In this case, samples are drawn uniformly from a d-dimensional hypersphere of unit radius, corresponding to a true log-volume of $\log V = d/2\log(\pi) - \log\Gamma(d/2+1)$.

Figure 5 shows that both `HBOXhull` and `PCABOXhull` are inaccurate, with errors increasing rapidly as dimensionality grows. The convex hull (`Chull`) is accurate for small d, but its performance deteriorates quickly in higher dimensions. Among kernel-based methods, only `HypSVM` perform reasonably well, although it requires large sample sizes to remain competitive. The GMM-based estimators remain accurate and stable, particularly when using the automatically selected threshold, achieving MAPE values below 10% up to $d = 20$. This behavior parallels that of the ellipsoidal hull estimator (`Ehull`), which is naturally well suited to this data-generating geometry. Overall, `GMMhull[LHS]` and `GMMhull[IS]` exhibit nearly identical performance, with the most notable difference arising from threshold selection. The data-driven threshold systematically reduces bias and improves precision relative to the fixed threshold case, reinforcing the support and practical value of the proposed selection procedure.

3.4 | Hyperellipsoid Scenario

Here, data are generated from a uniform distribution within a correlated d-dimensional hyperellipsoid. The analytic log-volume can be computed as $\log V = d/2\log(\pi) - \log\Gamma(d/2+1) + 0.5\log(\det(\boldsymbol{A}))$, where $\boldsymbol{A}$ is the $d \times d$ shape matrix with Toeplitz-type correlation structure:

$$\mathbf{A} = \begin{pmatrix} 1 & 0.8 & 0.8^2 & \cdots & 0.8^{d-1} \\ 0.8 & 1 & 0.8 & \cdots & 0.8^{d-2} \\ 0.8^2 & 0.8 & 1 & \cdots & 0.8^{d-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0.8^{d-1} & 0.8^{d-2} & 0.8^{d-3} & \cdots & 1 \end{pmatrix}$$

Figure 6 confirms that traditional geometric estimators, such as `HBOXhull` and `PCABOXhull`, provides very poor estimates of the log-volume in this setting. Both `Chull` and `HypSVM` perform reasonable well only at the smallest dimensionality ($d = 2$), but their accuracy deteriorates rapidly as d increases. Consistent with the hypersphere case, the ellipsoidal hull estimator (`Ehull`) is the most accurate competitor, as it aligns with the underlying data-generating geometry. The GMM-based estimators, `GMMhull[LHS]` and `GMMhull[IS]`, follow closely and maintain good accuracy across all dimensions. Notably, for the highest-dimensional configuration ($d = 20$), the importance sampling (IS) approach yields significant better performance than LHS, reflecting its improved efficiency in high-dimensional spaces. Moreover, the automatically selected threshold provides systematically more accurate estimates than the fixed value case, underscoring the benefits of the data-driven thresholding strategy.

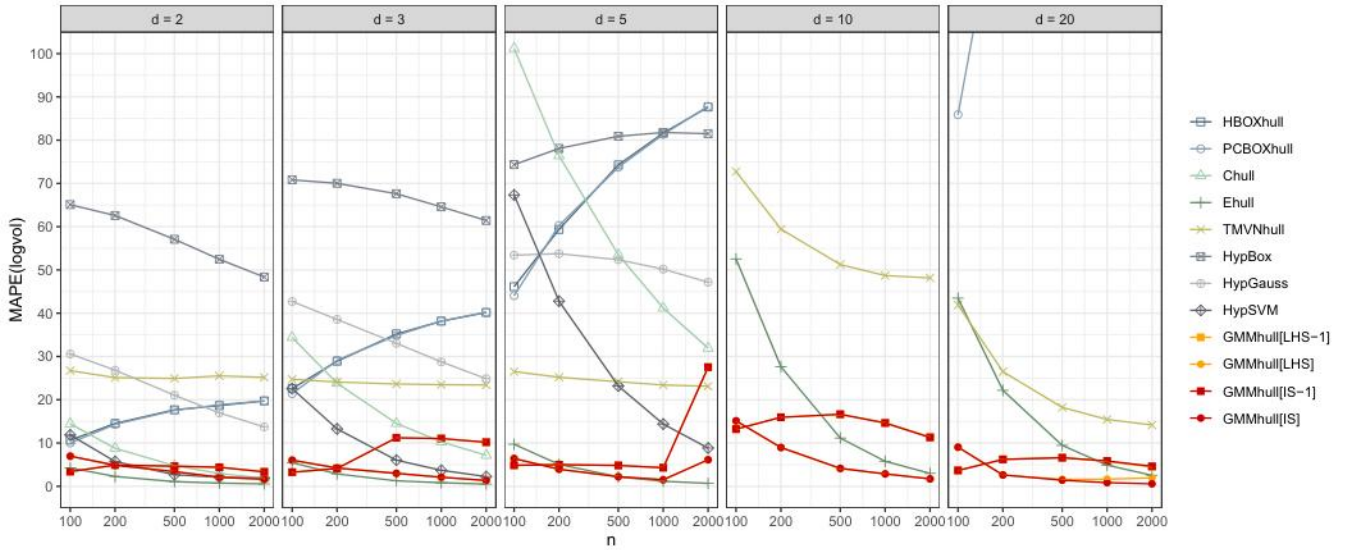

FIGURE 5 | Mean absolute percentage error (MAPE) of log-volume estimates for datasets uniformly distributed within a d-dimensional hypersphere. Results are reported for varying sample sizes (n) and dimensions (d). The upper limit of the y-axis is truncated at 100%, so some estimators are not shown for configurations where their error exceeds this threshold.

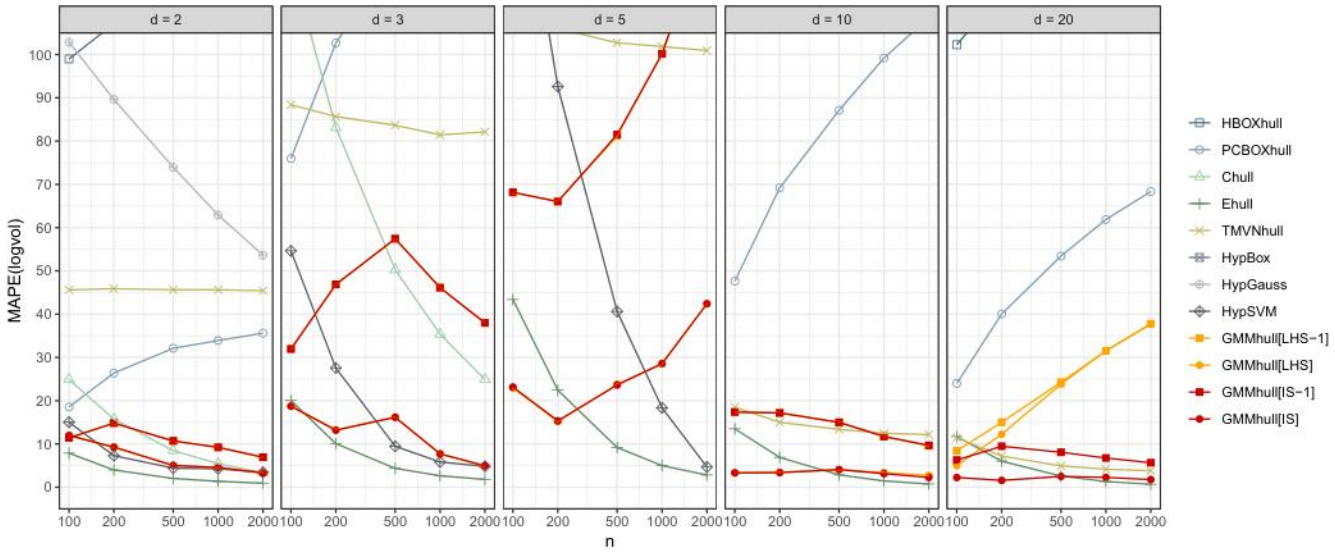

FIGURE 6 | Mean absolute percentage error (MAPE) of log-volume estimates for datasets uniformly distributed within a d-dimensional hyperellipsoid. Results are reported for varying sample sizes (n) and dimensions (d). The upper limit of the y-axis is truncated at 100%, so some estimators are not shown for configurations where their error exceeds this threshold.

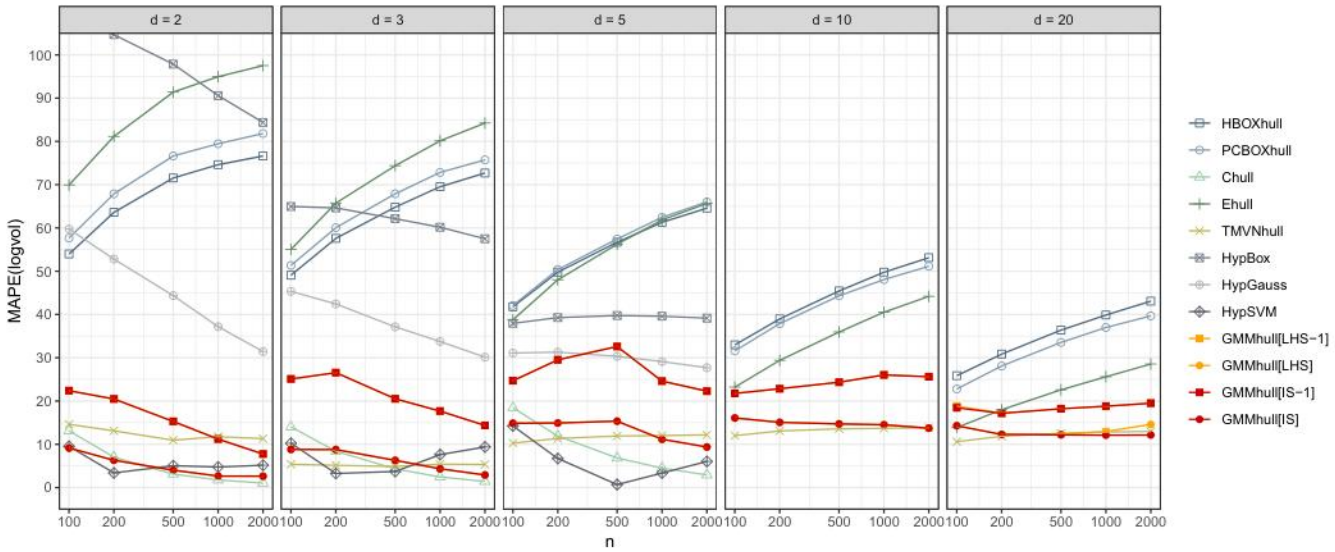

FIGURE 7 | Mean absolute percentage error (MAPE) of log-volume estimates for datasets uniformly distributed within a regular d-dimensional simplex with unit side length. Results are reported for varying sample sizes (n) and dimensions (d). The upper limit of the y-axis is truncated at 100%, so some estimators are not shown for configurations where their error exceeds this threshold.

3.5 | Simplex Scenario

In this case, data are drawn from a regular d-dimensional simplex with unit side length. The corresponding log-volume is $\log V = 0.5\log(d+1) - d/2\log(2) - \log\Gamma(d+1)$.

Figure 7 shows that, among competitors, only the truncated multivariate normal estimator (`TMVNhull`), the convex hull (`Chull`), and the SVM-based estimator (`HypSVM`) yield reasonably accurate log-volume estimates. All remaining estimators struggle to capture the simplex boundary accurately, resulting in substantial bias. However, both `Chull` and `HypSVM` become infeasible as dimensionality increases, a limitation already observed in the previous scenarios. In contrast, the GMM-based estimators using importance sampling and the adaptive threshold selection maintain reasonable accuracy even in high-dimensional settings, with performance improving as the sample size increases.

3.6 | Overlapping Hyperspheres Scenario

This scenario involves a mixture of two partially overlapping hyperspheres with equal weights, simulating a multimodal non-elliptical distribution. The first hypersphere is generated with mean vector $\boldsymbol{\mu}_1 = [0.5/\sqrt{d}, \ldots, 0.5/\sqrt{d}]^\top$ and

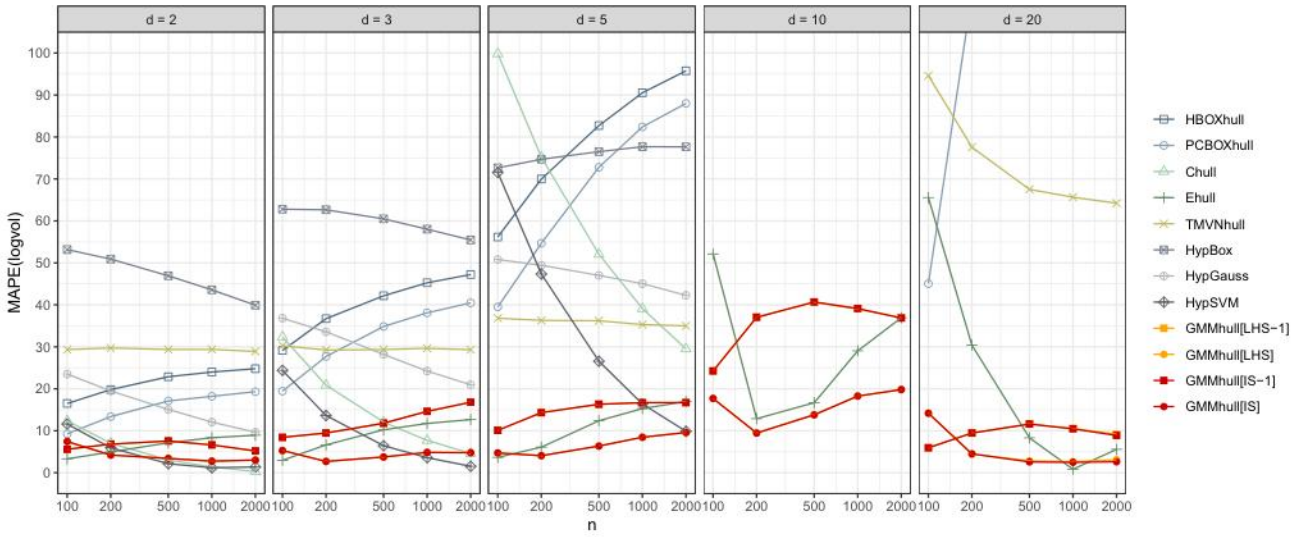

FIGURE 8 | Mean absolute percentage error (MAPE) of log-volume estimates for datasets uniformly generated from two overlapping d-dimensional hyperspheres of different sizes. The upper limit of the y-axis is truncated at 100%, so some estimators are not shown for configurations where their error exceeds this threshold.

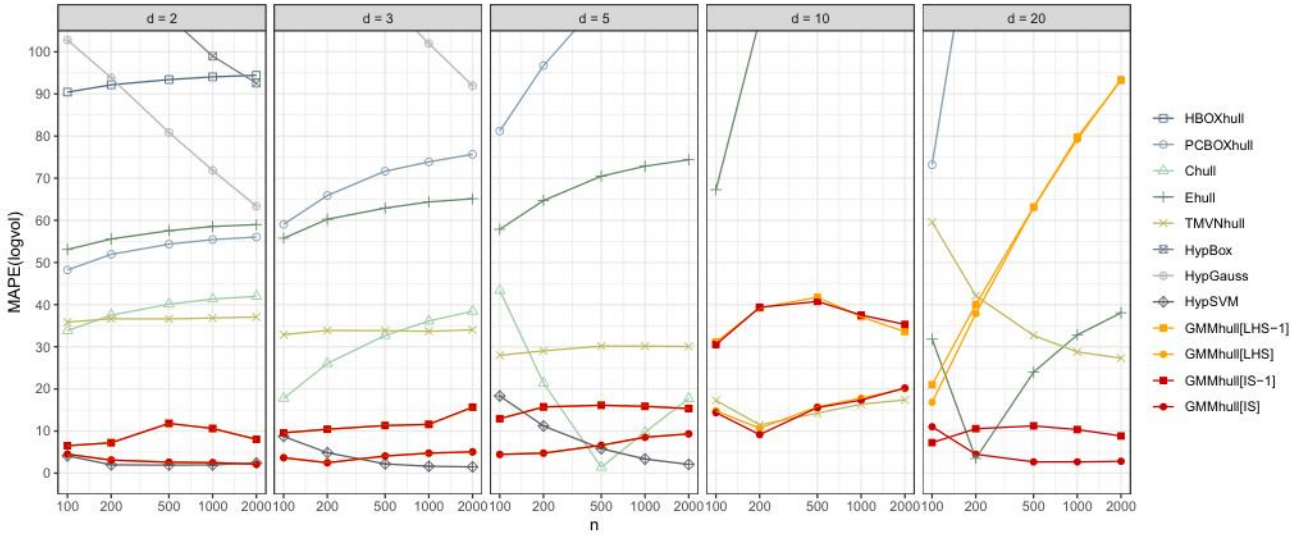

FIGURE 9 | Mean absolute percentage error (MAPE) of log-volume estimates for datasets uniformly generated from two separated d-dimensional hyperspheres of different sizes. The upper limit of the y-axis is truncated at 100%, so some estimators are not shown for configurations where their error exceeds this threshold.

shape matrix $\boldsymbol{A}_1 = 0.5\boldsymbol{I}_d$, while the second is centered at $\boldsymbol{\mu}_2 = [-0.5/\sqrt{d}, \ldots, -0.5/\sqrt{d}]^\top$ and shape matrix $\boldsymbol{A}_2 = \boldsymbol{I}_d$. Hence, the two components differ in both location and scale. The true log-volume can be obtained by adding the analytical volumes of the two hyperspheres and subtracting their intersection, the latter computed via a high-precision Monte Carlo simulation.

As shown in Figure 8, both `HBOXhull` and `PCBOXhull` fail to provide reliable volume estimates as dimensionality increases, with errors that worsen when the sample size becomes large. Kernel-based estimators also struggle: while `HypSVM` achieves reasonable accuracy in low dimensions, it requires very large sample sizes to remain competitive in higher dimensions, and the remaining kernel estimators perform poorly across all configurations. Most competing methods degrade substantially with increasing dimension, with the notable exception of `Ehull` and the GMM-based estimators using the automatically selected threshold. The adaptive GMM-based estimators remain accurate and stable even in high-dimensional settings and clearly outperform their counterparts that rely on a fixed threshold level. This highlights, once again, the key role played by the proposed threshold selection strategy.

3.7 | Separated Hyperspheres Scenario

The final scenario is a modification of the previous case, obtained by increasing the separation between the two hyperspheres. In particular, the first hypersphere is generated with mean vector $\boldsymbol{\mu}_1 = [2/\sqrt{d}, \ldots, 2/\sqrt{d}]^\top$ and shape matrix $\boldsymbol{A}_1 = 0.5\boldsymbol{I}_d$, while

the second is centered at $\boldsymbol{\mu}_2 = [-2/\sqrt{d}, \ldots, -2/\sqrt{d}]^\top$ with shape matrix $\boldsymbol{A}_2 = \boldsymbol{I}_d$. The resulting hyperspheres are well separated, so the true log-volume can be simply obtained by summing the analytical volumes of the two hyperspheres.

Based on the results shown in Figure 9, the only competitors to GMM-based estimators are `HypSVM` in low dimensions, and to a lesser extent, `Ehull` in high dimensions. The GMM-based estimator using importance sampling and automatically selected threshold, `GMMhull[IS]`, achieves consistently higher accuracy, with improvements that are even more evident than in the overlapping hypersphere scenario. Moreover, in high dimensions, the advantage of importance sampling over Latin Hypercube Sampling appears striking.

3.8 | Discussion

Across all experimental configurations, the GMM-based hypervolume estimator demonstrates consistently good performance in terms of accuracy, flexibility, and scalability compared with competing methods. Simple estimators based on hyperrectangles, such as `HBOXhull` and `PCBOXhull`, or hyperellipsoids, such as `Ehull` and `TMVNhull`, are highly inaccurate unless their geometry is aligned with the underlying data-generating mechanism. Other traditional estimators, including `Chull`, perform adequately only in low-dimensional settings but deteriorate rapidly as dimensionality or shape complexity increases. Among the kernel-based estimators, only the SVM-based method (`HypSVM`) achieves reasonable accuracy, yet it remains feasible only in low dimensions and requires large sample sizes. In contrast, the GMM-based framework provides stable and accurate performance across all considered scenarios.

A key factor influencing the quality of the GMM-based estimator of the hypervolume is the choice of the threshold level h. The comparison between fixed and automatically selected threshold reveals systematic and substantial differences across all simulation studies. When the threshold is fixed at the value corresponding to a GMM-hull that encloses all data points, the resulting estimates generally exhibit larger values of MAPE. By contrast, the data-driven selection procedure based on the HDR change-point criterion discussed in Section 2.2.3 produces threshold values that consistently reduce MAPE across all dimensions and data shapes. The choice of sampling scheme has a comparatively smaller impact on accuracy, although importance sampling (IS) tends to yield more accurate estimates under specific circumstances. In particular, when the region occupied by the data represents a tiny fraction of the empirical hyperrectangle $\mathcal{H}$, as often occurs in higher dimensions, for elongated or disconnected structures, or at very high HDR levels, IS yields substantial improvements.

Overall, the proposed GMM-based hypervolume estimation framework exhibits good and consistent accuracy across different data-generating mechanisms, sample sizes, and dimensionalities. The automatic threshold selection procedure adopted is a critical component of its performance, allowing the estimator to adapt to the underlying density structure and thereby improving reliability in both unimodal and multimodal high-dimensional contexts. The importance sampling variant further amplifies these benefits, maintaining high accuracy with superior computational efficiency in the most challenging high-dimensional settings.

4 | Applications

This section illustrates the practical use of the proposed GMM-based hypervolume estimator in two applied domains: anomaly detection and ecological niche estimation. These examples complement the simulation study by demonstrating how the estimator behaves when used as a component of broader statistical procedures, including noise detection in mixture models and the identification of multivariate niche regions in ecological data.

4.1 | Anomaly Detection

Anomaly detection aims at identifying observations that deviate substantially from the expected structure of a dataset. Such anomalous points, often referred to as outliers or noise, may signal important events such as sensor failures, fraud, measurement errors, or other unusual conditions. Anomaly detection methods are widely applied across many fields including finance, cybersecurity, healthcare, manufacturing monitoring, and environmental science.

A principled approach to anomaly detection was proposed by [2], who augmented a Gaussian mixture model with a uniform noise component:

$$f(\boldsymbol{x}) = \sum_{k=1}^{G} \pi_k \phi(\boldsymbol{x}|\boldsymbol{\mu}_k, \Sigma_k) + \pi_0 \frac{1}{V}, \tag{6}$$

where V denotes the hypervolume of the data region, and $\pi_k \geq 0$ are the mixing proportions satisfying $\sum_{k=0}^{G} \pi_k = 1$.

The effectiveness of this approach requires a reasonable initial set of candidate noise points and an accurate estimate of V. Several strategies are described in [17, sec. 7.1]. The GMM-based hypervolume estimator introduced in Section 2.2 enables both tasks to be carried out in a data-driven manner using the following procedure:

1. Compute the estimate $\widehat{V}_{\text{GMM}}$ of the hypervolume by selecting the threshold $h = \min\left(\left\{\widehat{f}(\boldsymbol{x}_i)\right\}_{i=1}^{n}\right)$, where $\widehat{f}(\boldsymbol{x}_i)$ is the density of i-th data point estimated using (4). This corresponds to using the 100% HDR, so the resulting GMM hull is the region enclosing all observed data points, including possible outliers.
2. Assuming a uniform distribution for the noise component over the identified GMM hull, the density of each data point within this region is equal to $1/\widehat{V}_{\text{GMM}}$. Therefore, a preliminary noise allocation is made by defining the set

$$\tilde{C}_0 = \left\{\boldsymbol{x}_i : \widehat{f}(\boldsymbol{x}_i) < 1/\widehat{V}_{\text{GMM}}, \text{ for } i = 1, \ldots, n\right\},$$

with size $\tilde{n}_0 = |\tilde{C}_0|$.

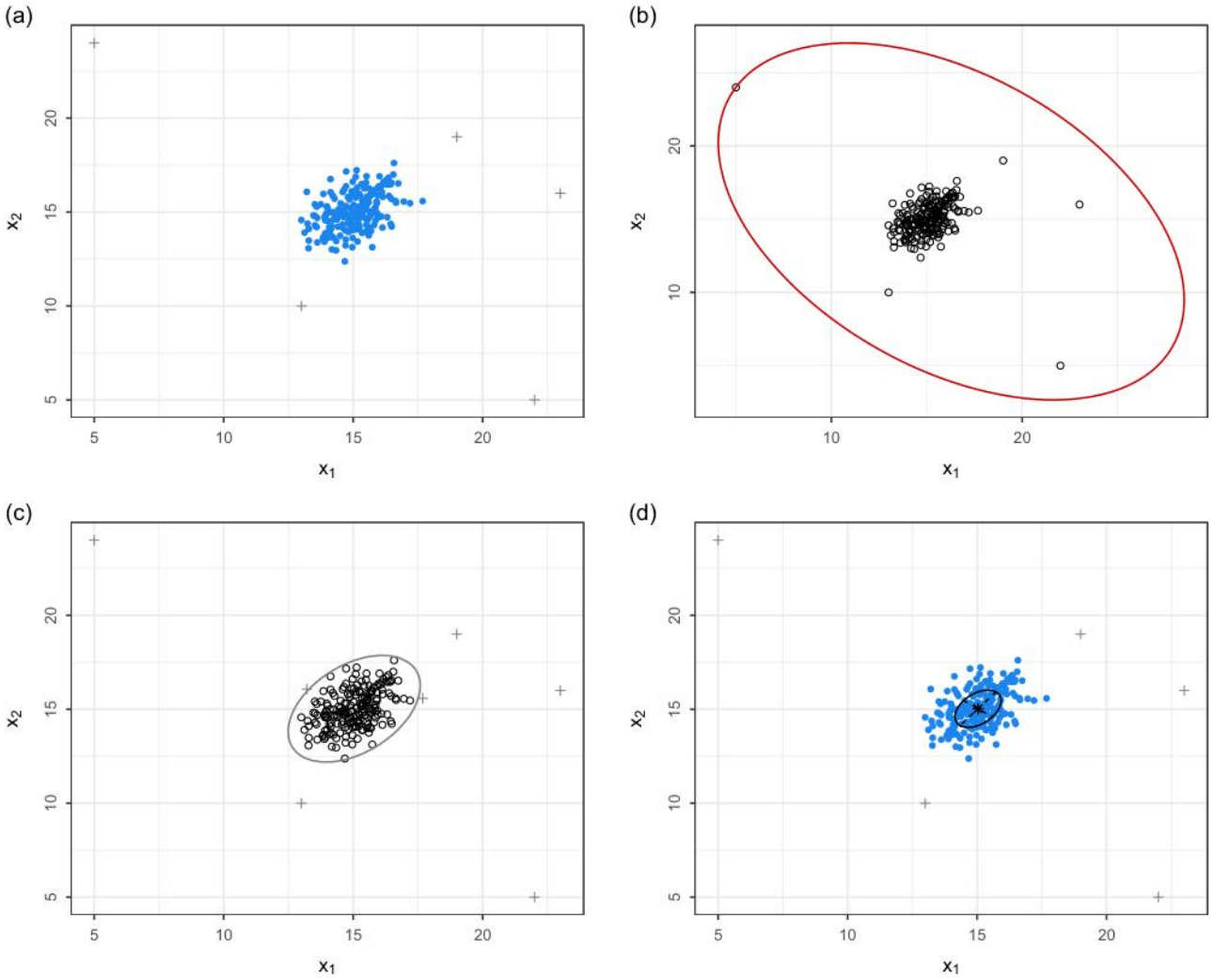

FIGURE 10 | Simulated data from a single Gaussian component with a few outliers. (a) Scatterplot of the generated data, with outliers marked as +. (b) GMM hull at 100% highest-density region (HDR). (c) Initial noise allocation of points having density smaller than $1/\widehat{V}_{\text{GMM}}$. (d) Final clustering after EM estimation of model in (6), showing the main Gaussian cluster and the detected outliers.

3. Allocate the remaining $n - \tilde{n}_0$ observations using, for instance, model-based agglomerative clustering [32, 33] to obtain an initial partition

$$\tilde{C}_k = \left\{ \mathbf{x}_i : \widehat{f}(\mathbf{x}_i) \geq 1/\widehat{V}_{\text{GMM}} \wedge \mathbf{x}_i \in \mathcal{P}_k, \quad \text{for } i = 1, \ldots, n \right\},$$

where $\mathcal{P}_k$ is the k-th subset of the partition with size $\tilde{n}_k = |\tilde{C}_k|$ for $k = 1, \ldots, G$, so that $n = \tilde{n}_0 + \tilde{n}_1 + \ldots + \tilde{n}_G$.

4. Fit model (6) via the EM algorithm using the initial partition $\left\{\tilde{C}_0, \tilde{C}_1, \ldots, \tilde{C}_G\right\}$ and the estimated volume $\widehat{V}_{\text{GMM}}$ to start the algorithm.

5. Output the final clusters $C_1, \ldots, C_G$ and the set of data anomalies C_0, such that $\left\{C_0 \cup C_1 \cup \ldots \cup C_G\right\} = \left\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\right\}$ and $C_j \cap C_k = \emptyset$ for $j \neq k$.

For volume estimation, the importance sampling (IS) approach is used. In low-dimensional settings, Latin hypercube sampling (LHS) would yield comparable results.

4.1.1 | Simulated Data: Single Gaussian Distribution With Outliers

Consider the synthetic dataset in Figure 10 a, showing a sample generated from a single Gaussian component with a few outliers added on the outskirts. Figure 10b shows the GMM hull obtained by setting the density threshold to the minimum fitted density, ensuring that the resulting contour encloses all observed data points. Figure 10c illustrates the initial noise allocation: the gray contour marks the region where $\widehat{f}(\mathbf{x}) > 1/\widehat{V}_{\text{GMM}}$, representing the bulk of the data, while the low-density points lying outside this region are allocated to the initial noise component. The initial partition succeeds in separating most outliers, then the EM fitting of model in (6) refines the classification and correctly identify all anomalous points, as shown in Figure 10d.

4.1.2 | Simulated Data: Three-Component Gaussian Mixture With Uniform Random Noise

A more challenging scenario is shown in Figure 11a, where data arise from a three-component Gaussian mixture contaminated

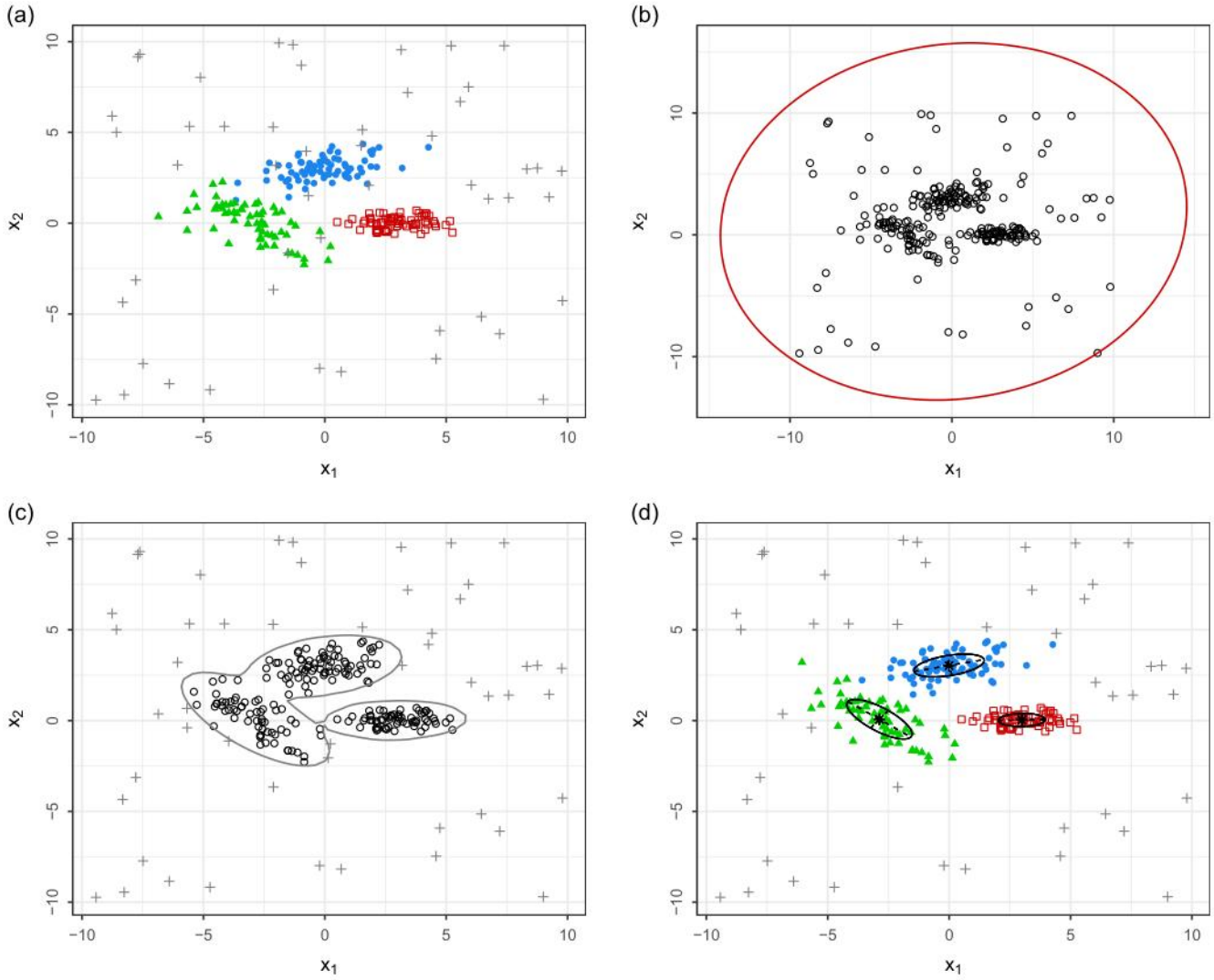

FIGURE 11 | Simulated data from a three-component Gaussian mixture with uniform random noise. (a) Scatterplot of the generated data, with outliers marked as +. (b) GMM hull at 100% highest-density region (HDR). (c) Initial noise allocation of points having density smaller than $1/\widehat{V}_{\text{GMM}}$. (d) Final clustering after EM estimation of model in (6), showing the main clusters and the detected outliers.

by uniformly distributed noise data points. The uniform noise creates density plateaus that would be difficult to detect without an explicit hypervolume estimate. The GMM-based hypervolume acts as a data-driven estimator, enabling the initial uniform component to be identified as the set of points below density $1/\widehat{V}_{\text{GMM}}$. After model (6) is estimated via EM, the fitted clusters correctly recover the three Gaussian components while separating the uniform outliers almost perfectly.

4.1.3 | Data Application: Minefields

The minefields dataset provided by Dasgupta and Raftery [2], shown in Figure 12a, consists of background clutter data points interspersed with a small set of observations arranged along an arrow-shaped pattern corresponding to buried mines. The GMM-based hypervolume estimator provides a principled way to initiate the noise-versus-signal separation. Figure 12b shows the GMM hull obtained from the fitted mixture model, capturing the broad extent of the clutter. In Figure 12c, points whose density falls below the threshold $1/\widehat{V}_{\text{GMM}}$ are flagged as potential anomalies, a set that includes the arrow-shaped mine pattern, but also some isolated groups of data points. After EM fitting of model (6), the final allocation cleanly separates the structured minefield signature from the background clutter (see Figure 12d).

4.2 | Ecological Niche Estimation

In ecological studies, niche estimation based on stable isotopes measured in muscle tissue provides a powerful approach for quantifying an organism's trophic position and resource use over time [34]. Stable isotope ratios (typically of carbon, nitrogen, and sulfur) in muscle tissue reflect the integrated

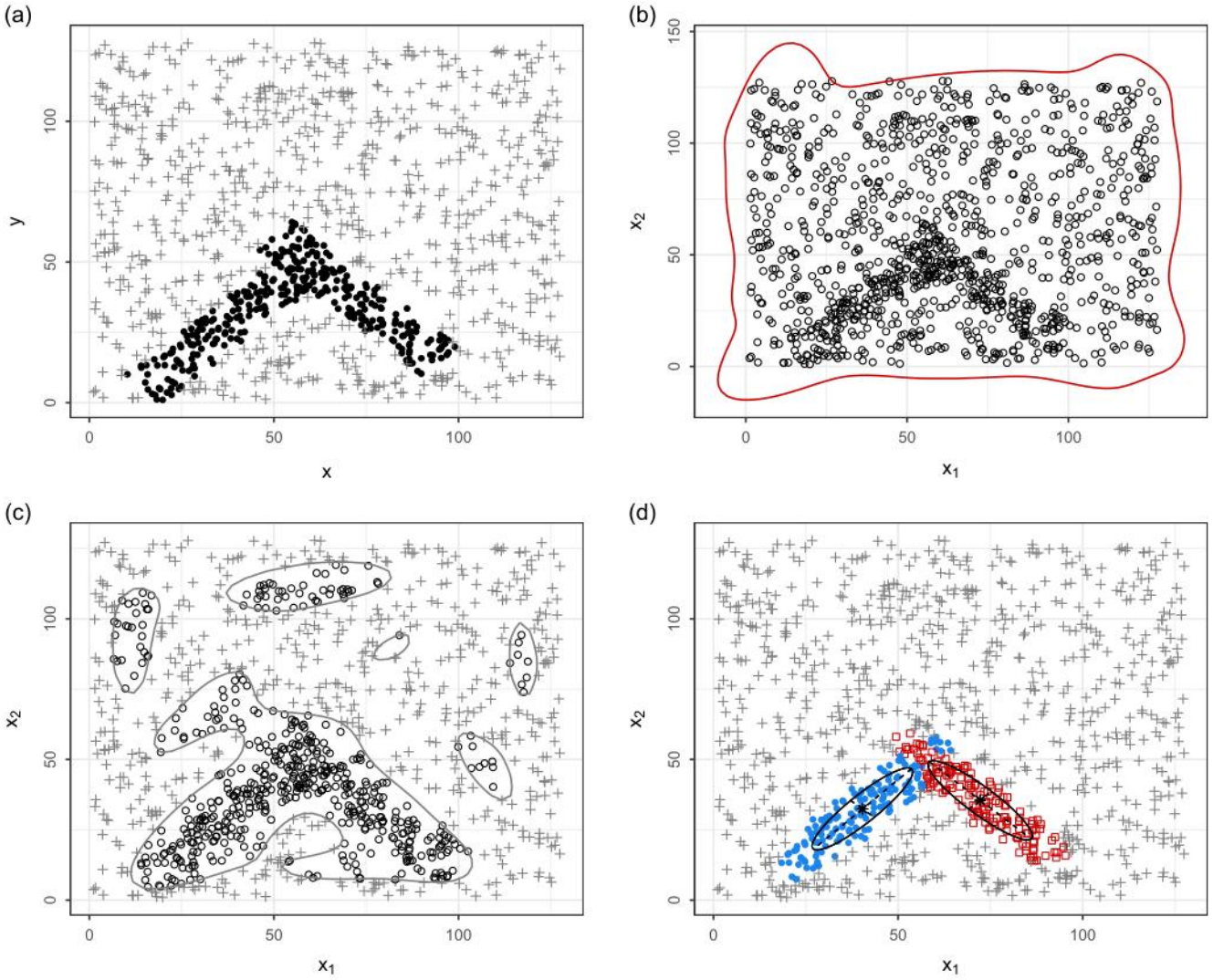

FIGURE 12 | Anomaly detection example based on the minefields dataset. (a) Observed data consisting of clutter points (+) with a structured arrow-shaped pattern corresponding to buried mines (•). (b) GMM hull corresponding to the 100% HDR region (red contour), which encloses the full extent of the clutter and mine pattern. (c) Initial noise allocation of points having density smaller than $1/\widehat{V}_{\mathrm{GMM}}$. (d) Final clustering after EM estimation of model (6), cleanly separating the structured mine pattern from the background clutter.

dietary signal accumulated over the period of tissue turnover. By analyzing these isotope values, researchers can infer the types and sources of food consumed, characterize the isotopic niche of a species or population, and assess niche overlap among species or groups.

TABLE 1 | Estimated hypervolumes quantifying niche size for the four Arctic fish species, derived from the GMM–based niche hulls.

	ARCS	BDWF	LKWF	LSCS
Niche size	90.34	1126.91	420.07	218.57

4.2.1 | Arctic Fish Species

The `fish` dataset, included in the `nicheROVER` R package [27], contains measurements of three stable isotopes—δ^{15} N (nitrogen), δ^{13} C (carbon), and δ^{34} S (sulfur)—obtained from the muscle tissue of four species of Arctic fish. The dataset comprises 278 individual observations, each representing a single fish and reporting its species identity along with the corresponding isotope values. It serves as a representative example for estimating and comparing trophic niche regions and quantifying niche overlap within a multivariate isotopic space. Such analyses are fundamental for understanding resource use, dietary segregation, and potential competition among closely related species coexisting within the same ecological community. The four fish species included in the dataset are Arctic Cisco (*Coregonus autumnalis*, ARCS), Broad Whitefish (*Coregonus nasus*, BDWF), Lake Whitefish (*Coregonus clupeaformis*, LKWF), and Least Cisco (*Coregonus sardinella*, LSCS).

Figures B1–B4 in Appendix B show the distributions of the three isotopes measured in the muscle tissue for the four Arctic fish species, along with the estimated GMM hulls. Following the procedure described in Section 2.2, the size of each niche region is quantified by its estimated hypervolume.

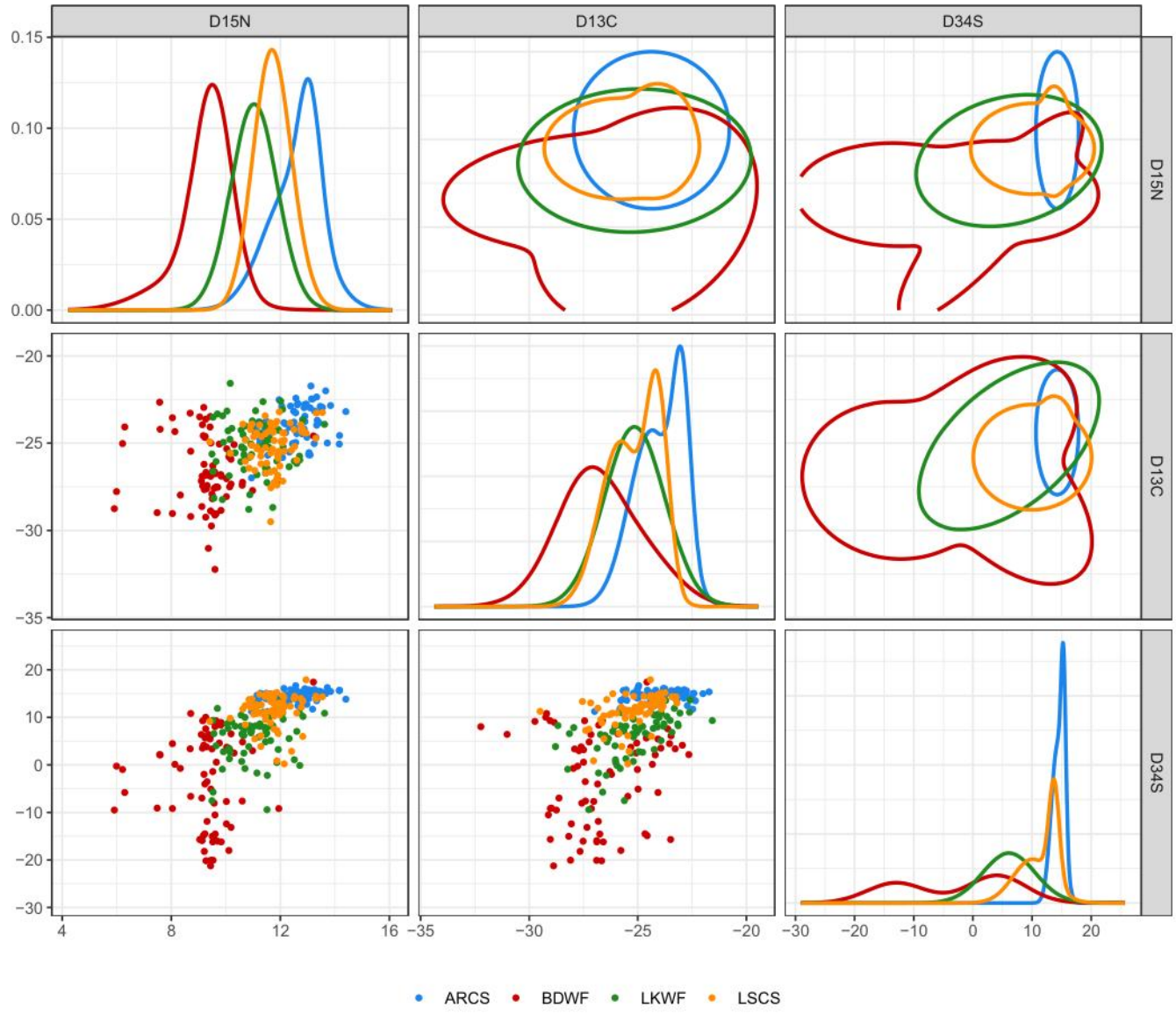

FIGURE 13 | GMM–based niche hulls estimated for the four Arctic fish species, showing differences in niche size and overlap among species.

Table 1 reports the resulting hypervolumes estimated using importance sampling (IS) for each species. The results indicate substantial differences in niche size among the four Arctic fish species, with *Coregonus autumnalis* (ARCS) exhibiting the smallest niche region and *Coregonus nasus* (BDWF) the largest. See also Figure 13.

Following Swanson et al. [9], pairwise niche overlap can be quantified as the probability that a subject from one species falls within the niche region of another. Accordingly, the probability that an observation from Species A is found within the niche of Species B can be expressed as follows:

$$\Pr(\mathbf{x} \in \mathcal{H}_B | \mathbf{x} \in \mathcal{H}_A) = \mathbb{E}_{f_A}\left[\mathbb{1}(\mathbf{x} \in \mathcal{H}_B)\right].$$

where $\mathcal{H}_A$ and $\mathcal{H}_B$ are the GMM-based niche hulls for species A and B, respectively. This can be estimated using the following Monte Carlo sampling procedure:

1. Simulate S random data points $\{\tilde{\mathbf{x}}_s\}_{s=1}^{S}$ from the GMM estimated for Species A;
2. Compute the density values $\{f_B(\tilde{\mathbf{x}}_s)\}_{s=1}^{S}$ from the GMM estimated for Species B;
3. Estimate the pairwise niche overlap as the probability of Species A individuals falling within the GMM hull of Species B by computing the proportion of simulated points with density above the HDR density threshold for Species B, that is,

$$O(B|A) = \frac{1}{S}\sum_{s=1}^{S}\mathbb{1}\left(f_B(\tilde{\mathbf{x}}_s) \geq h_B\right).$$

Table 2 reports the estimated pairwise niche overlaps (in percentage) among the four Arctic fish species. Notice that overlap values are not symmetric. For instance, the value O(LSCS|BDWF) $\approx$ 62% indicates that *Coregonus sardinella* (LSCS) lies largely within

TABLE 2 | Pairwise niche overlap among the four Arctic fish species based on the estimated GMM hulls.

O(col|row)	ARCS	BDWF	LKWF	LSCS
ARCS		30.95	77.64	72.98
BDWF	0.75		31.89	4.91
LKWF	9.88	74.02		63.10
LSCS	54.95	62.11	95.23	

Note: The overlap $O($ col.$|$ row $)$ indicates the probability (%) that an observation from the species in the row falls within the niche region of the species in the column.

the broad trophic niche of *Coregonus nasus* (BDWF). Conversely, O(BDWF|LSCS) $\approx$ 5% shows that only a very small fraction of BDWF individuals fall within the niche region of LSCS, highlighting the asymmetry and the marked difference in niche breadth between the two species.

Finally, Figure 13 provides a visualization of the estimated trophic niche regions in multivariate isotopic space, showing clear differences in location, volume, and shape across Arctic fish species. The GMM-based approach yields smoothly varying niche boundaries and accommodates multimodal or elongated structures that would be difficult to capture with other approaches.

5 | Discussion and Conclusions

The hypervolume represents the multidimensional extent of a dataset, quantifying the region in feature space occupied by its observations. Hypervolume estimation plays a central role across diverse domains, and several methods have been developed for this purpose. However, despite their popularity, existing approaches face notable limitations, particularly in high dimensions or when the occupied region exhibits complex structures. These difficulties arise from several structural features of the underlying distribution, including multimodality, low-density regions connecting clusters, elongated or irregular boundaries, disconnected components, and tail behavior, all of which may strongly affect the geometry of the resulting density level set. These challenges motivate the development of more flexible and computationally efficient strategies.

In this paper, we have proposed a novel methodology for estimating the hypervolume of multivariate datasets based on finite Gaussian mixtures. The resulting mixture-based framework provides a flexible and probabilistic approach that adapts naturally to multimodal structures, irregular shapes, and heterogeneous data geometries. Through comparative analyses and applied examples, we demonstrated that the proposed estimator offers improved accuracy, robustness, and scalability relative to traditional geometric, kernel, and convex-hull-based alternatives. It is worth noting that over- and under-estimation of the log-hypervolume may carry different practical implications depending on the application context, and investigating this asymmetry is left as a direction for future research.

Building on the contributions of this paper, several avenues for future research appear promising. One possible direction is to extend the framework to incorporate non-Gaussian mixture components, such as skewed or heavy-tailed distributions, which would broaden applicability to datasets with pronounced asymmetry or heavier tails. Additionally, exploration of dimension-reduction strategies that preserve hypervolume-relevant structures, sequential or online estimation procedures for streaming data, and accelerated Monte Carlo techniques for large-scale or high-dimensional settings could further enhance the practicality and computational efficiency of the methodology. Finally, linking the proposed framework to uncertainty quantification, for instance by deriving confidence regions for the estimated hypervolume, would support rigorous inference in scientific applications.

Acknowledgments

I would like to express my gratitude to the Istituto Nazionale di Fisica Nucleare (INFN) for providing the scientific computing infrastructure used for running the simulations presented in this work. Open access publishing facilitated by Universita di Bologna, as part of the Wiley - CRUI-CARE agreement.

Funding

The author has nothing to report.

Conflicts of Interest

The author declares no conflicts of interest.

Data Availability Statement

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

1. G. Hutchinson, "Concluding Remarks," *Cold Spring Harbor Symposia on Quantitative Biology* 22 (1957): 415–427.

2. A. Dasgupta and A. E. Raftery, "Detecting Features in Spatial Point Processes With Clutter via Model-Based Clustering," *Journal of the American Statistical Association* 93, no. 441 (1998): 294–302, https://doi.org/10.1080/01621459.1998.10474110.

3. A. P. Guerreiro, C. M. Fonseca, and L. Paquete, "The Hypervolume Indicator: Computational Problems and Algorithms," *ACM Computing Surveys* 54, no. 6 (2021): 1–42, https://doi.org/10.1145/3453474.

4. M. Metodiev, M. Perrot-Dockès, S. Ouadah, N. J. Irons, P. Latouche, and A. E. Raftery, "Easily Computed Marginal Likelihoods From Posterior Simulation Using the THAMES Estimator," *Bayesian Analysis* 20, no. 3 (2025): 1003–1030, https://doi.org/10.1214/24-BA1422.

5. F. P. Preparata and M. I. Shamos, *Computational Geometry: An Introduction* (Springer Science & Business Media, 2012).

6. C. B. Barber, D. P. Dobkin, and H. Huhdanpaa, "The Quickhull Algorithm for Convex Hulls," *ACM Transactions on Mathematical Software* 22, no. 4 (1996): 469–483.

7. B. Blonder, C. Lamanna, C. Violle, and B. J. Enquist, "The *n*-Dimensional Hypervolume," *Global Ecology and Biogeography* 23, no. 5 (2014): 595–609, https://doi.org/10.1111/geb.12146.

8. B. Blonder, C. B. Morrow, B. Maitner, et al., "New Approaches for Delineating n-Dimensional Hypervolumes," *Methods in Ecology and Evolution* 9, no. 2 (2018): 305–319, https://doi.org/10.1111/2041-210X.12865.

9. H. K. Swanson, M. Lysy, M. Power, A. D. Stasko, J. D. Johnson, and J. D. Reist, "A New Probabilistic Method for Quantifying n-Dimensional

Ecological Niches and Niche Overlap," *Ecology* 96, no. 2 (2015): 318–324, https://doi.org/10.1890/14-0235.1.

10. S. G. Jarvis, P. A. Henrys, A. M. Keith, E. Mackay, S. E. Ward, and S. M. Smart, "Model-Based Hypervolumes for Complex Ecological Data," *Ecology* 100, no. 5 (2019): e02676, https://doi.org/10.1002/ecy.2676.

11. J. Bader and E. Zitzler, "HypE: An Algorithm for Fast Hypervolume-Based Many-Objective Optimization," *Evolutionary Computation* 19, no. 1 (2011): 45–76, https://doi.org/10.1162/EVCO_a_00009.

12. T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. (John Wiley & Sons, 2006).

13. G. J. McLachlan, S. X. Lee, and S. I. Rathnayake, "Finite mixture models," *Annual Review of Statistics and Its Application* 6, no. 1 (2019): 355–378, https://doi.org/10.1146/annurev-statistics-031017-100325.

14. G. J. McLachlan and D. Peel, *Finite Mixture Models* (Wiley, 2000).

15. J. Banfield and A. E. Raftery, "Model-Based Gaussian and Non-Gaussian Clustering," *Biometrics* 49 (1993): 803–821, https://doi.org/10.2307/2532201.

16. G. Celeux and G. Govaert, "Gaussian Parsimonious Clustering Models," *Pattern Recognition* 28 (1995): 781–793, https://doi.org/10.1016/0031-3203(94)00125-6.

17. L. Scrucca, C. Fraley, T. B. Murphy, and A. E. Raftery, *Model-Based Clustering, Classification, and Density Estimation Using Mclust in R* (Chapman & Hall/CRC, 2023).

18. C. Fraley and A. E. Raftery, "Bayesian Regularization for Normal Mixture Estimation and Model-Based Clustering," *Journal of Classification* 24, no. 2 (2007): 155–181, https://doi.org/10.1007/s00357-007-0004-5.

19. C. Keribin, "Consistent Estimation of the Order of Mixture Models," *Sankhya Series A* 62, no. 1 (2000): 49–66.

20. G. Schwarz, "Estimating the Dimension of a Model," *Annals of Statistics* 6, no. 2 (1978): 461–464, https://doi.org/10.1214/aos/1176344136.

21. A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum Likelihood From Incomplete Data via the EM Algorithm (With Discussion)," *Journal of the Royal Statistical Society, Series B: Statistical Methodology* 39 (1977): 1–38, https://doi.org/10.1111/j.2517-6161.1977.tb01600.x.

22. M. D. McKay, R. J. Beckman, and W. J. Conover, "Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output From a Computer Code," *Technometrics* 21, no. 2 (1979): 239–245, https://doi.org/10.1080/00401706.1979.10489755.

23. A. B. Owen, "A Central Limit Theorem for Latin Hypercube Sampling," *Journal of the Royal Statistical Society: Series B: Methodological* 54, no. 2 (1992): 541–551, https://doi.org/10.1111/j.2517-6161.1992.tb01895.x.

24. M. Stein, "Large Sample Properties of Simulations Using Latin Hypercube Sampling," *Technometrics* 29, no. 2 (1987): 143–151, https://doi.org/10.1080/00401706.1987.10488205.

25. C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*, 2nd ed. (Springer, 2004).

26. R. J. Hyndman, "Computing and Graphing Highest Density Regions," *American Statistician* 50, no. 2 (1996): 120–126, https://doi.org/10.1080/00031305.1996.10474359.

27. M. Lysy, A. D. Stasko, and H. K. Swanson, "nicheROVER: Niche Region and Niche Overlap Metrics for Multidimensional Ecological Niches. R package version 1.1.2" (2023), https://doi.org/10.32614/CRAN.package.nicheROVER.

28. L. Scrucca, "mclustAddons: Addons for the 'mclust' Package. R package Version 0.10" (2025), https://doi.org/10.32614/CRAN.package.mclustAddons.

29. K. Habel, R. Grasman, R. B. Gramacy, P. Mozharovskyi, and D. C. Sterratt, "geometry: Mesh Generation and Surface Tessellation. R package version 0.5.2" (2025), https://doi.org/10.32614/CRAN.package.geometry.

30. M. Maechler, P. Rousseeuw, A. Struyf, M. Hubert, and K. Hornik, "cluster: Cluster Analysis Basics and Extensions. R package version 2.1.8.1" (2025), https://doi.org/10.32614/CRAN.package.cluster.

31. B. Blonder, C. B. Morrow, S. Brown, et al., "hypervolume: High Dimensional Geometry, Set Operations, Projection, and Inference Using Kernel Density Estimation, Support Vector Machines, and Convex Hulls. R package version 3.1.6" (2025), https://doi.org/10.32614/CRAN.package.hypervolume.

32. C. Fraley, "Algorithms for Model-Based Gaussian Hierarchical Clustering," *SIAM Journal on Scientific Computing* 20, no. 1 (1998): 270–281, https://doi.org/10.1137/S1064827596311451.

33. L. Scrucca and A. E. Raftery, "Improved Initialisation of Model-Based Clustering Using Gaussian Hierarchical Partitions," *Advances in Data Analysis and Classification* 4, no. 9 (2015): 447–460, https://doi.org/10.1007/s11634-015-0220-z.

34. S. Bearhop, C. E. Adams, S. Waldron, R. A. Fuller, and H. Macleod, "Determining Trophic Niche Width: A Novel Approach Using Stable Isotope Analysis," *Journal of Animal Ecology* 73, no. 5 (2004): 1007–1012, https://doi.org/10.1111/j.0021-8790.2004.00861.x.

Appendix A

Experimental Scenarios

In this appendix, graphical summaries of the synthetic two-dimensional datasets described in Section 3 are presented. For each experimental scenario, the simulated sample is shown alongside a selection of benchmark hypervolume estimators, including the data hyperrectangle, the principal-component hyperrectangle, the convex hull, the ellipsoidal hull, and the truncated multivariate normal hull. For comparison, the GMM-based estimator introduced in Section 2.2 is also displayed, with the GMM hulls constructed either by fixing the HDR level to 100% and using the automatically selected threshold based on the two-segment piecewise regression model discussed in Section 2.2.3. These visualizations complement the quantitative results reported in Section 3 by illustrating how the various estimators capture the underlying data geometry in a low-dimensional setting.

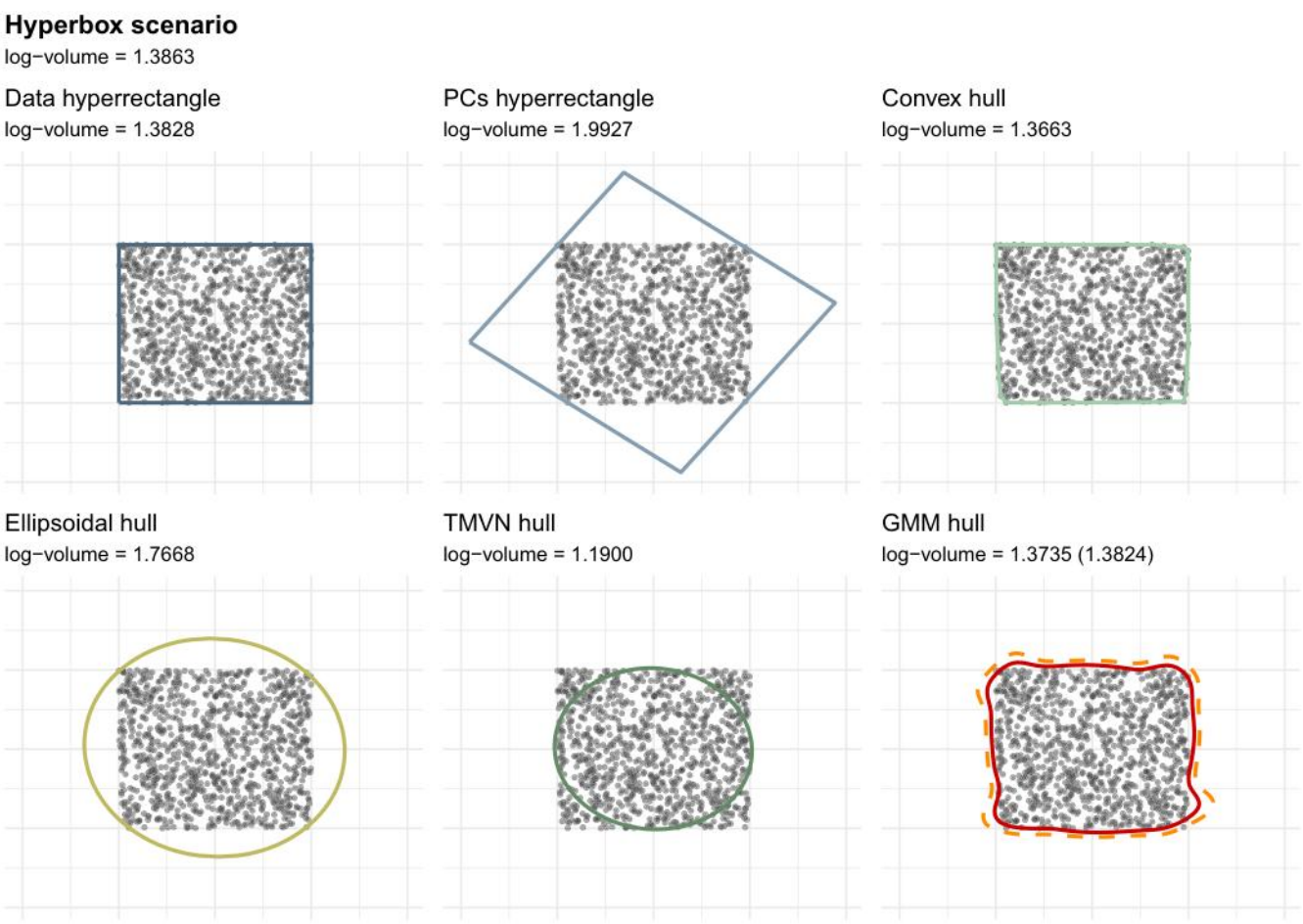

FIGURE A1 | Simulated two-dimensional dataset for the hyperbox scenario, displayed together with benchmark hypervolume estimators (data hyperrectangle, principal-component hyperrectangle, convex hull, ellipsoidal hull, and truncated multivariate normal hull) and the GMM-based estimator computed using importance sampling (IS). The latter is displayed in the bottom-right panel, showing the GMM hull at the fixed 100% HDR level (orange dashed contour; log-volume in parentheses) and at the automatically selected threshold obtained from the two-segment piecewise regression model (solid red contour; corresponding log-volume shown).

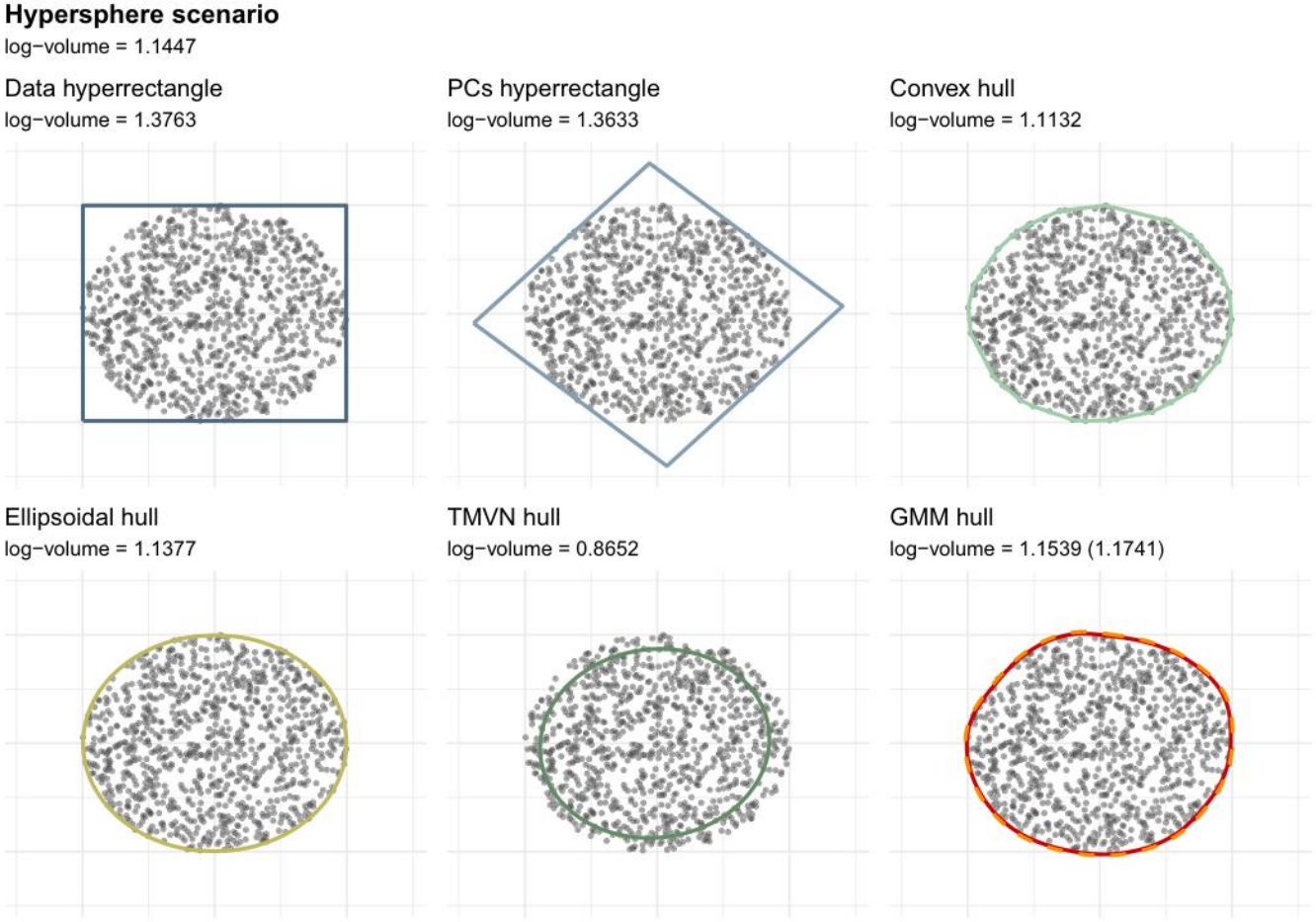

FIGURE A2 | Simulated two-dimensional dataset for the hypersphere scenario, displayed together with benchmark hypervolume estimators (data hyperrectangle, principal-component hyperrectangle, convex hull, ellipsoidal hull, and truncated multivariate normal hull) and the GMM-based estimator computed using importance sampling (IS). The latter is displayed in the bottom-right panel, showing the GMM hull at the fixed 100% HDR level (orange dashed contour; log-volume in parentheses) and at the automatically selected threshold obtained from the two-segment piecewise regression model (solid red contour; corresponding log-volume shown).

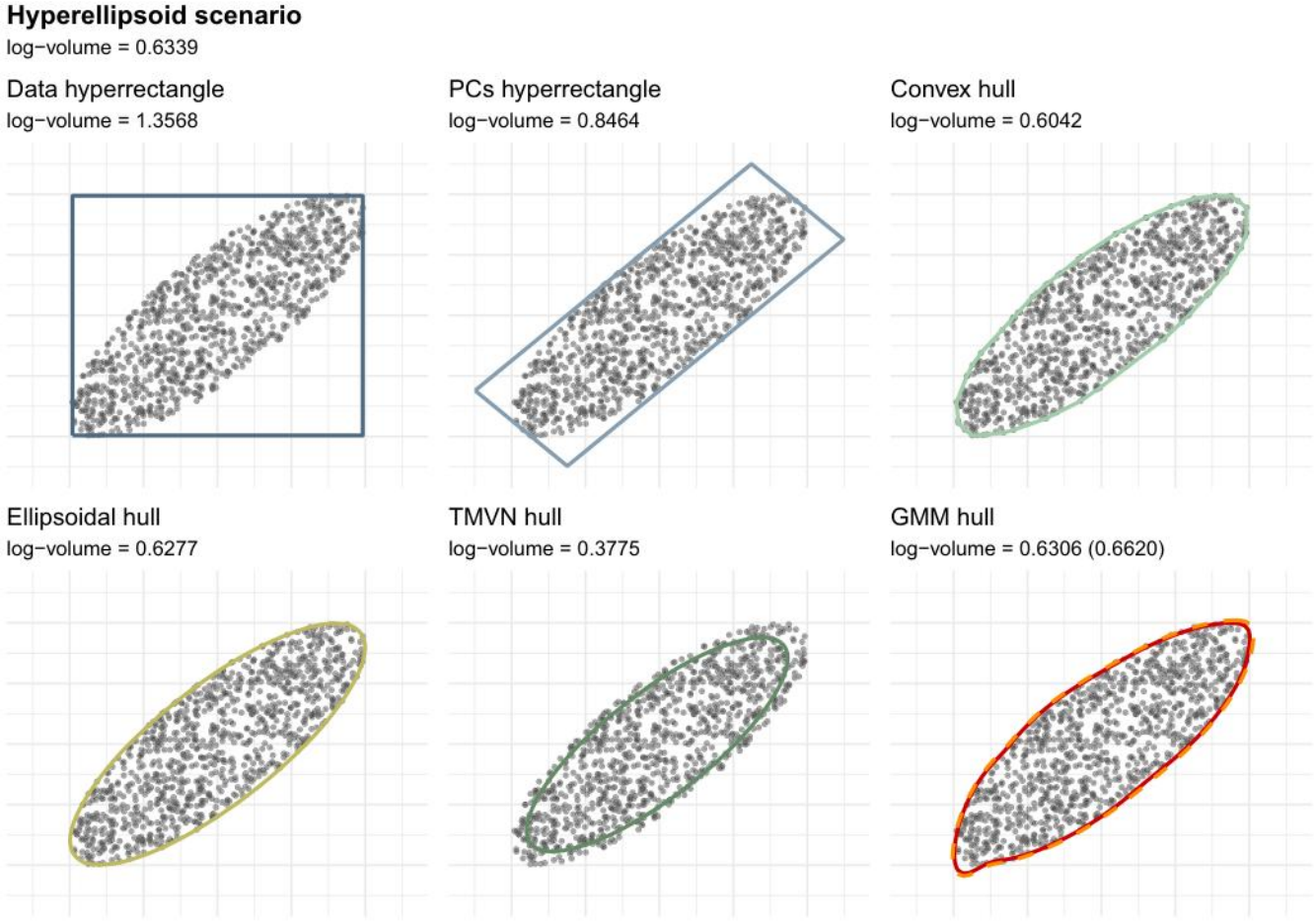

FIGURE A3 | Simulated two-dimensional dataset for the hyperellipsoid scenario, displayed together with benchmark hypervolume estimators (data hyperrectangle, principal-component hyperrectangle, convex hull, ellipsoidal hull, and truncated multivariate normal hull) and the GMM-based estimator computed using importance sampling (IS). The latter is displayed in the bottom-right panel, showing the GMM hull at the fixed 100% HDR level (orange dashed contour; log-volume in parentheses) and at the automatically selected threshold obtained from the two-segment piecewise regression model (solid red contour; corresponding log-volume shown).

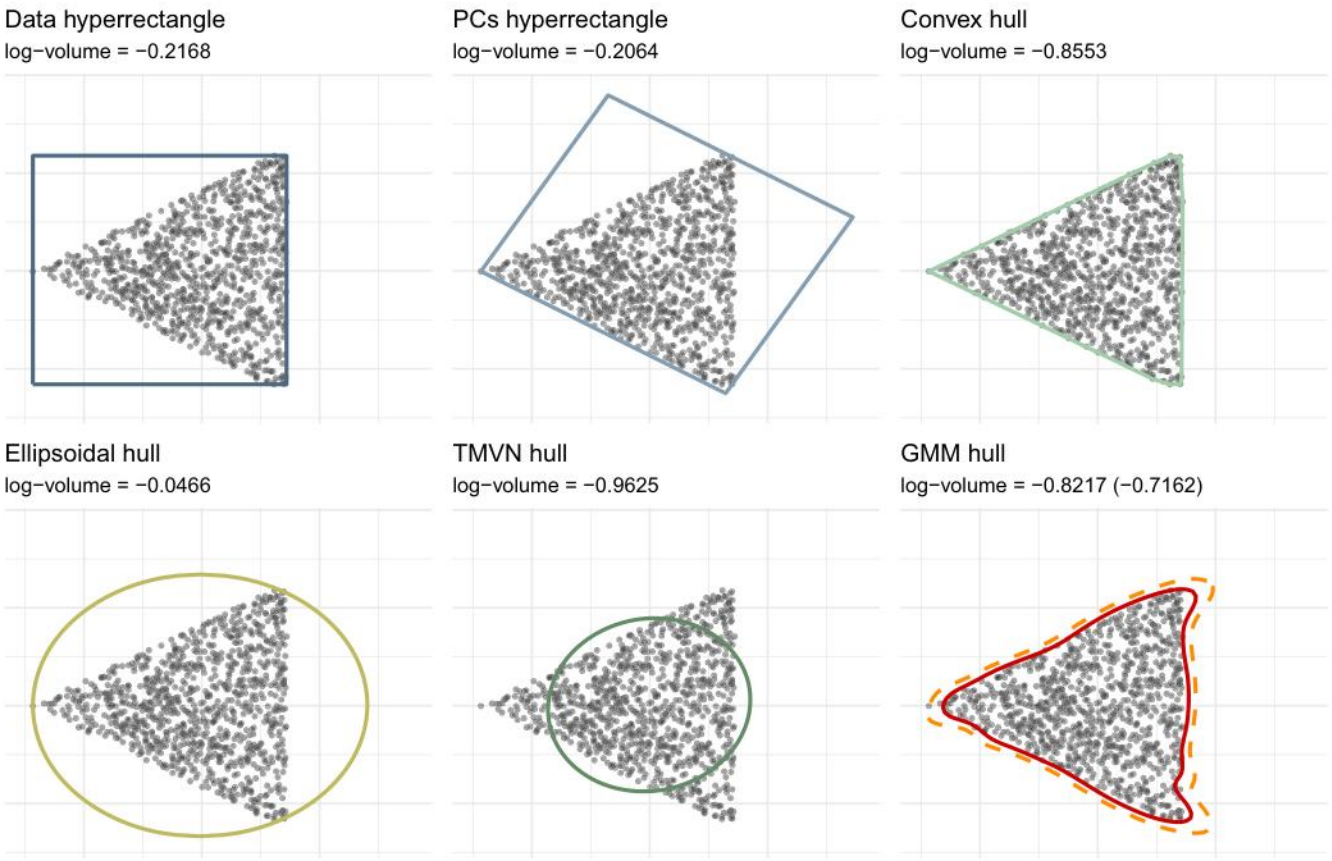

FIGURE A4 | Simulated two-dimensional dataset for the simplex scenario, displayed together with benchmark hypervolume estimators (data hyperrectangle, principal-component hyperrectangle, convex hull, ellipsoidal hull, and truncated multivariate normal hull) and the GMM-based estimator computed using importance sampling (IS). The latter is displayed in the bottom-right panel, showing the GMM hull at the fixed 100% HDR level (orange dashed contour; log-volume in parentheses) and at the automatically selected threshold obtained from the two-segment piecewise regression model (solid red contour; corresponding log-volume shown).

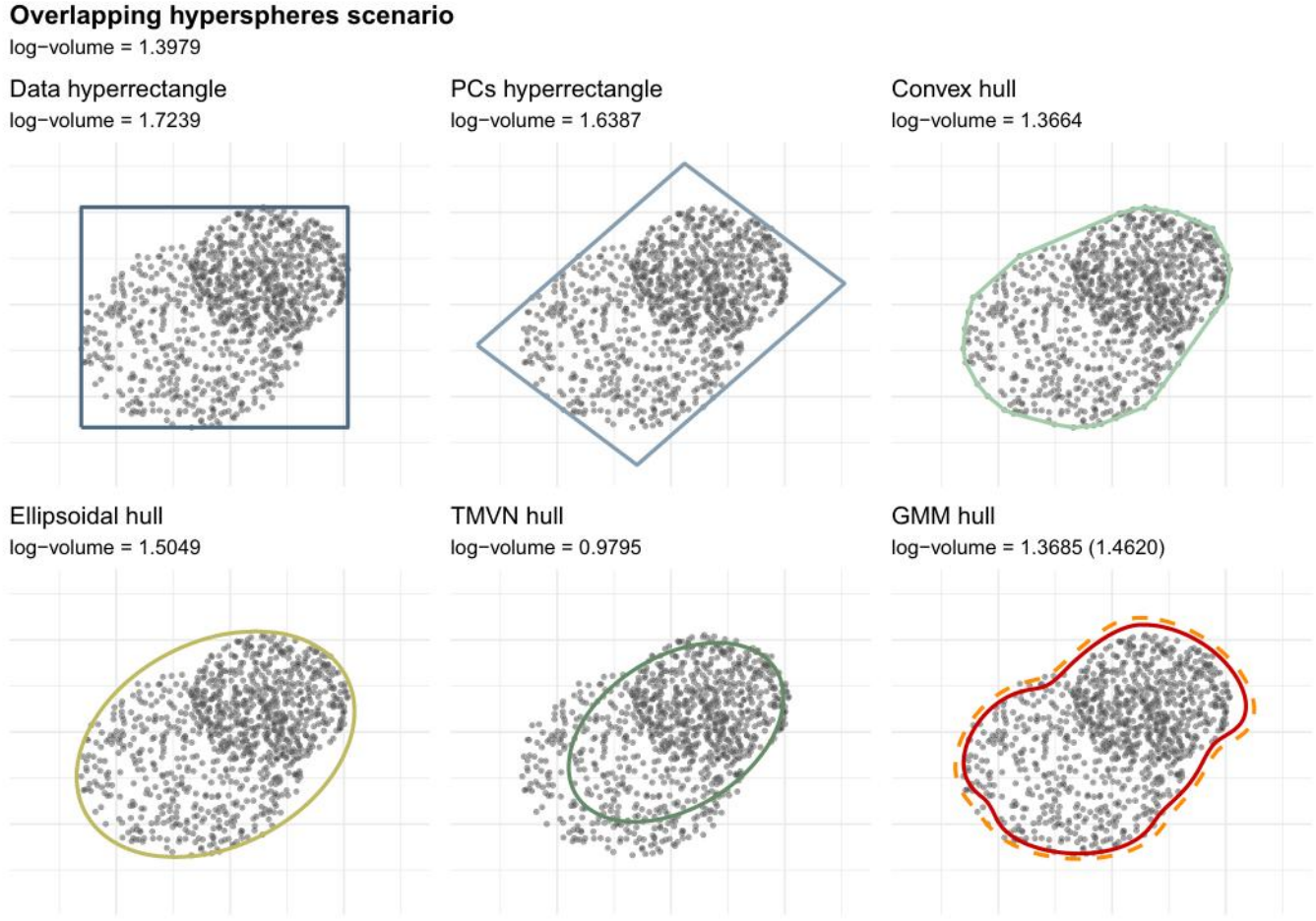

FIGURE A5 | Simulated two-dimensional dataset for the overlapping hyperspheres scenario, displayed together with benchmark hypervolume estimators (data hyperrectangle, principal-component hyperrectangle, convex hull, ellipsoidal hull, and truncated multivariate normal hull) and the GMM-based estimator computed using importance sampling (IS). The latter is displayed in the bottom-right panel, showing the GMM hull at the fixed 100% HDR level (orange dashed contour; log-volume in parentheses) and at the automatically selected threshold obtained from the two-segment piecewise regression model (solid red contour; corresponding log-volume shown).

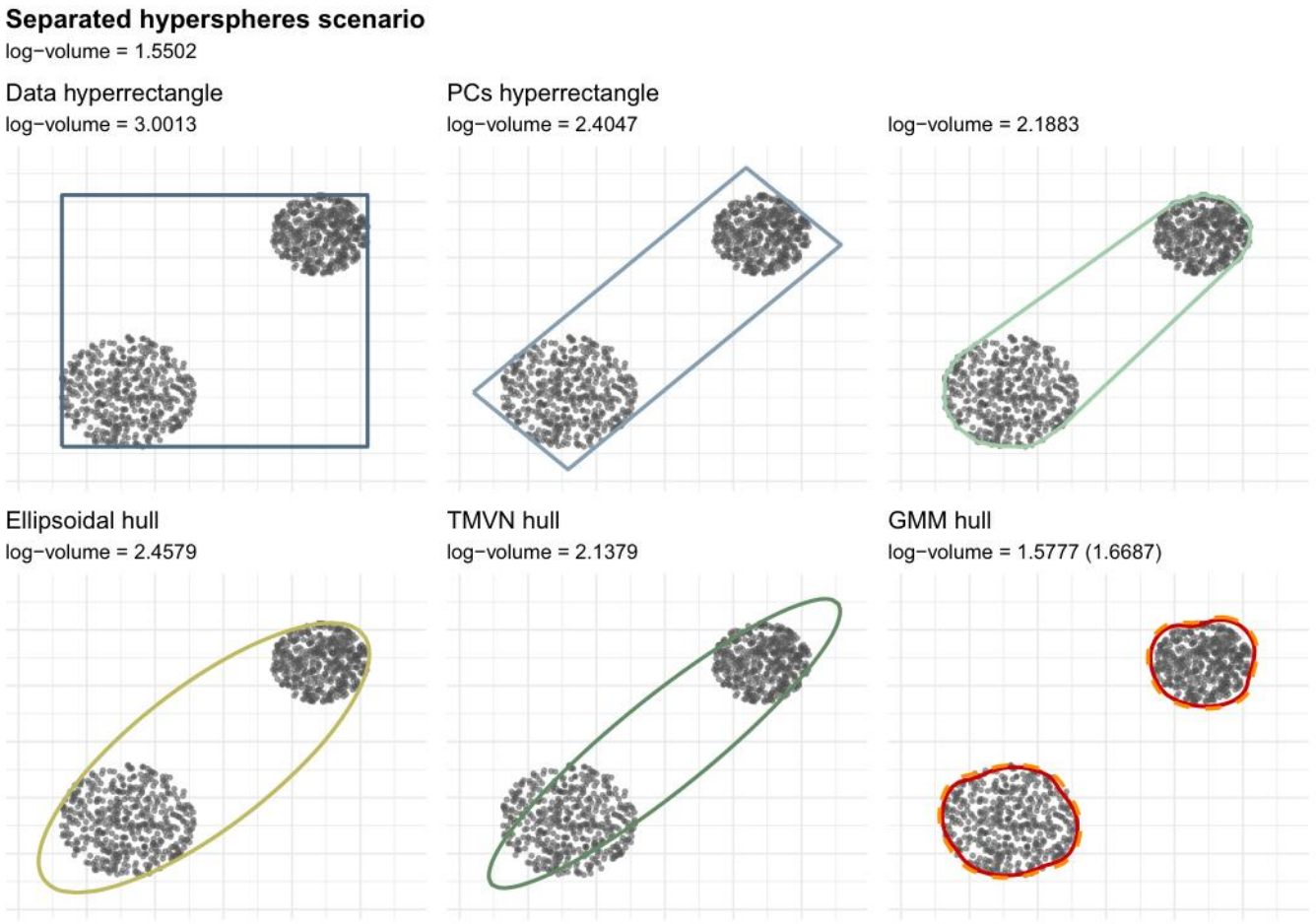

FIGURE A6 | Simulated two-dimensional dataset for the separated hyperspheres scenario, displayed together with benchmark hypervolume estimators (data hyperrectangle, principal-component hyperrectangle, convex hull, ellipsoidal hull, and truncated multivariate normal hull) and the GMM-based estimator computed using importance sampling (IS). The latter is displayed in the bottom-right panel, showing the GMM hull at the fixed 100% HDR level (orange dashed contour; log-volume in parentheses) and at the automatically selected threshold obtained from the two-segment piecewise regression model (solid red contour; corresponding log-volume shown).

Appendix B

Ecological Niche Estimation

In this appendix we report some graphical material accompanying the ecological niche estimation example discussed in Section 4.2. For each of the four Arctic fish species, the graphs show the distributions of the three stable isotopes measured in muscle tissue, together with the corresponding GMM-based niche hull. The figures illustrate how the proposed mixture-based approach captures the multivariate structure of each species' isotopic niche, including differences in location, dispersion, and shape.

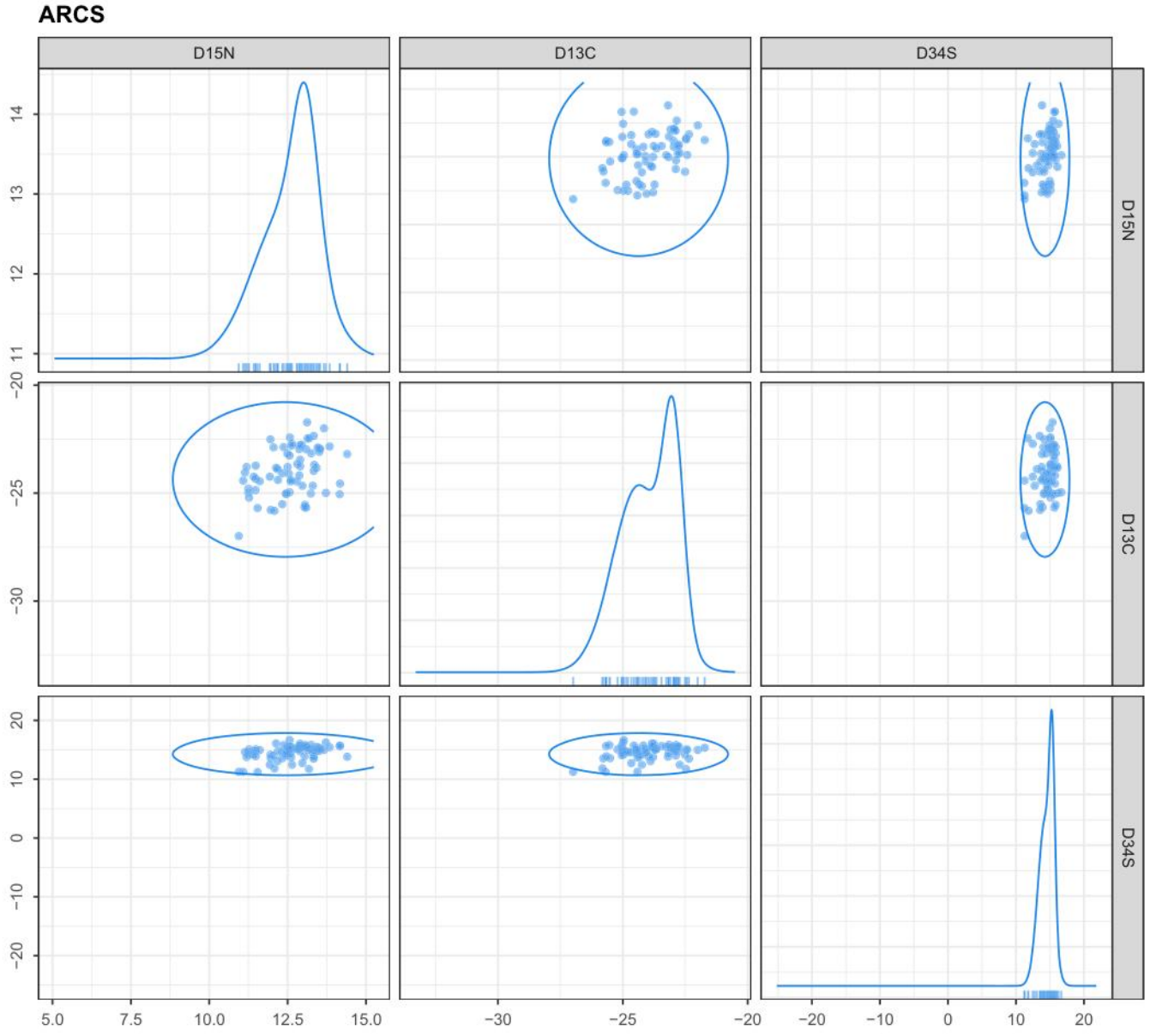

FIGURE B1 | Distribution of the three stable isotopes measured in the muscle tissue of *Coregonus autumnalis* (ARCS). Contours represent the GMM hull estimated according to the proposed method, delineating the trophic niche region of this species in multivariate isotopic space.

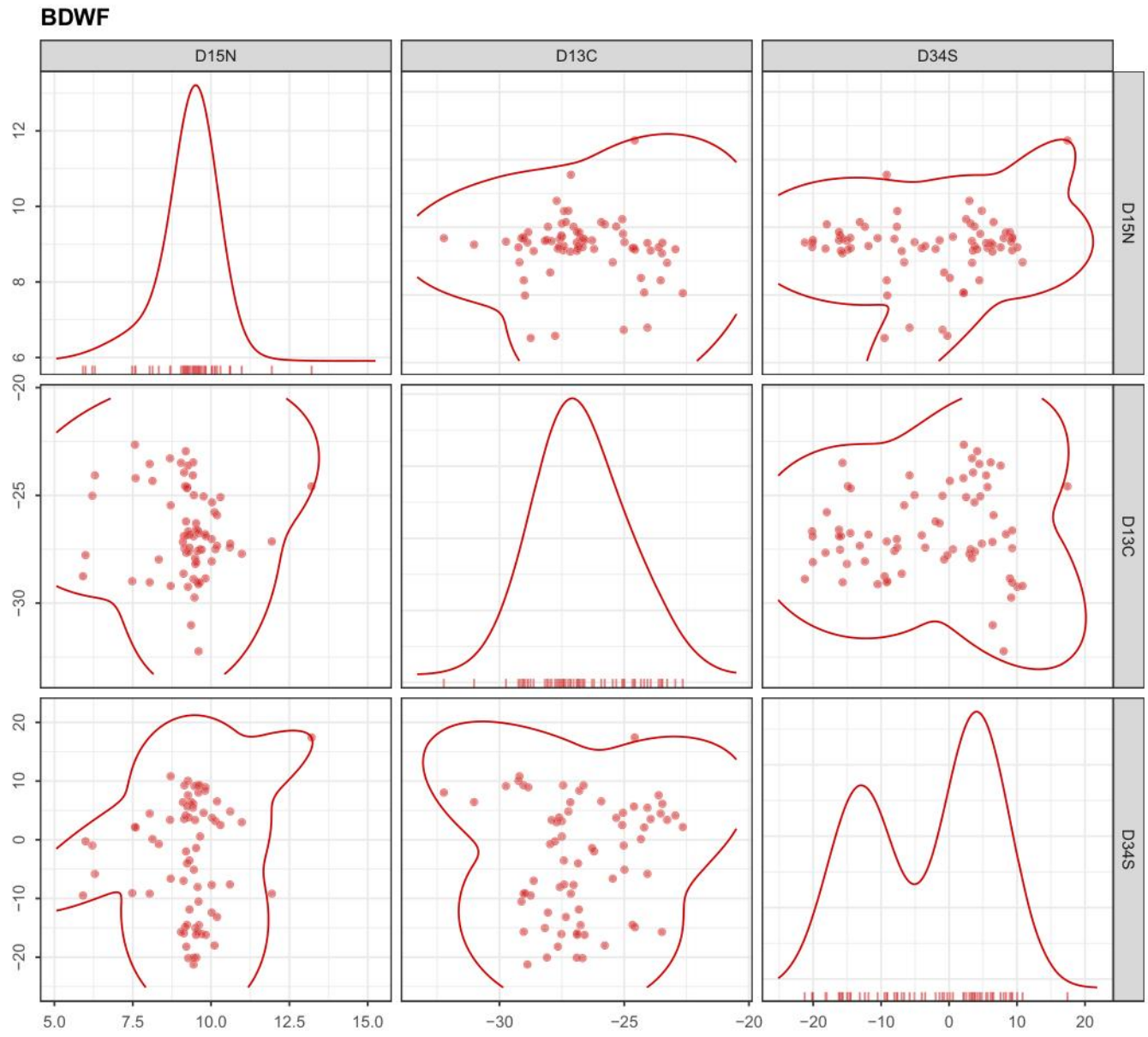

FIGURE B2 | Distribution of the three stable isotopes measured in the muscle tissue of *Coregonus nasus* (BDWF). Contours represent the GMM hull estimated according to the proposed method, delineating the trophic niche region of this species in multivariate isotopic space.

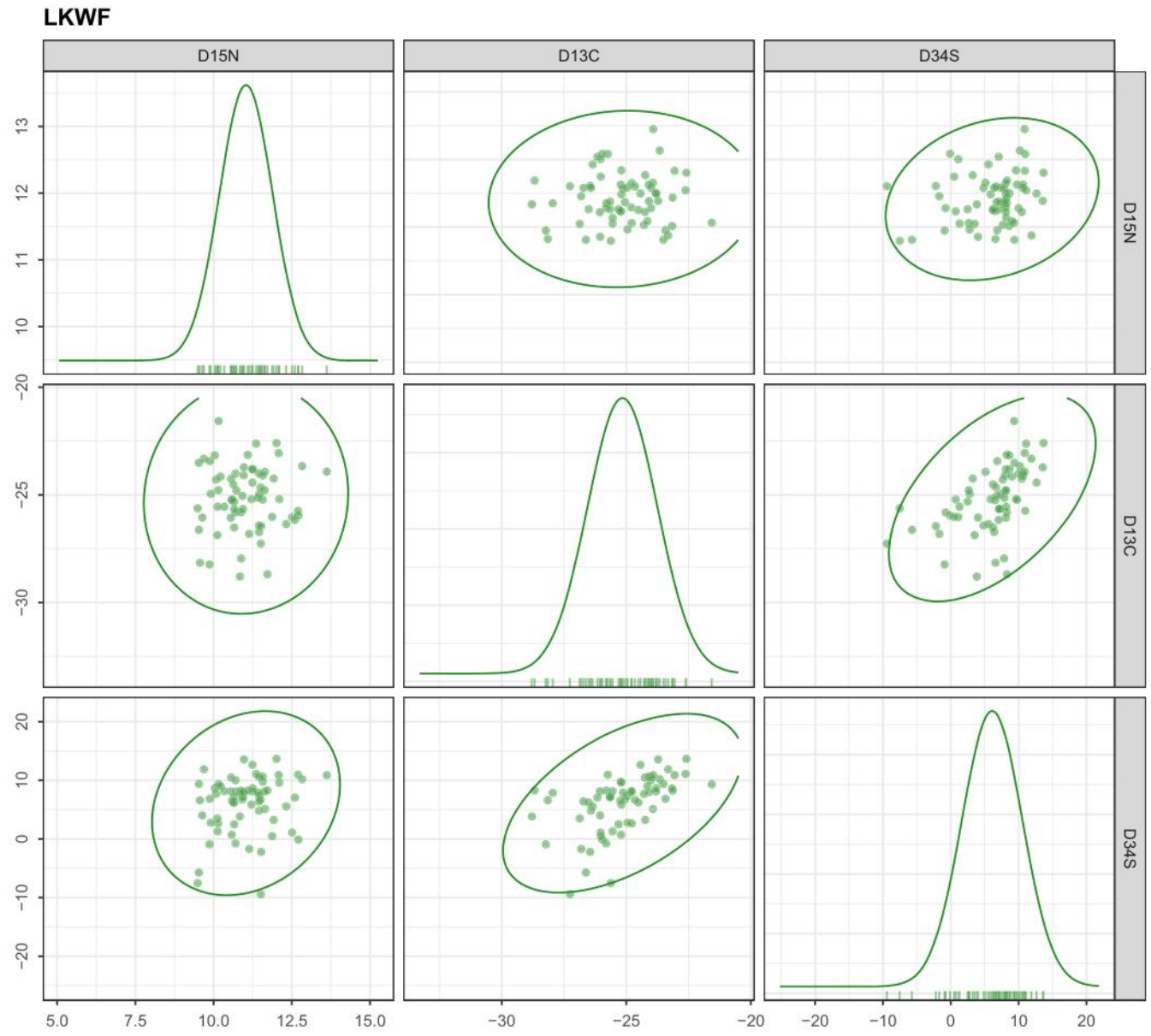

FIGURE B3 | Distribution of the three stable isotopes measured in the muscle tissue of *Coregonus clupeaformis* (LKWF). Contours represent the GMM hull estimated according to the proposed method, delineating the trophic niche region of this species in multivariate isotopic space.

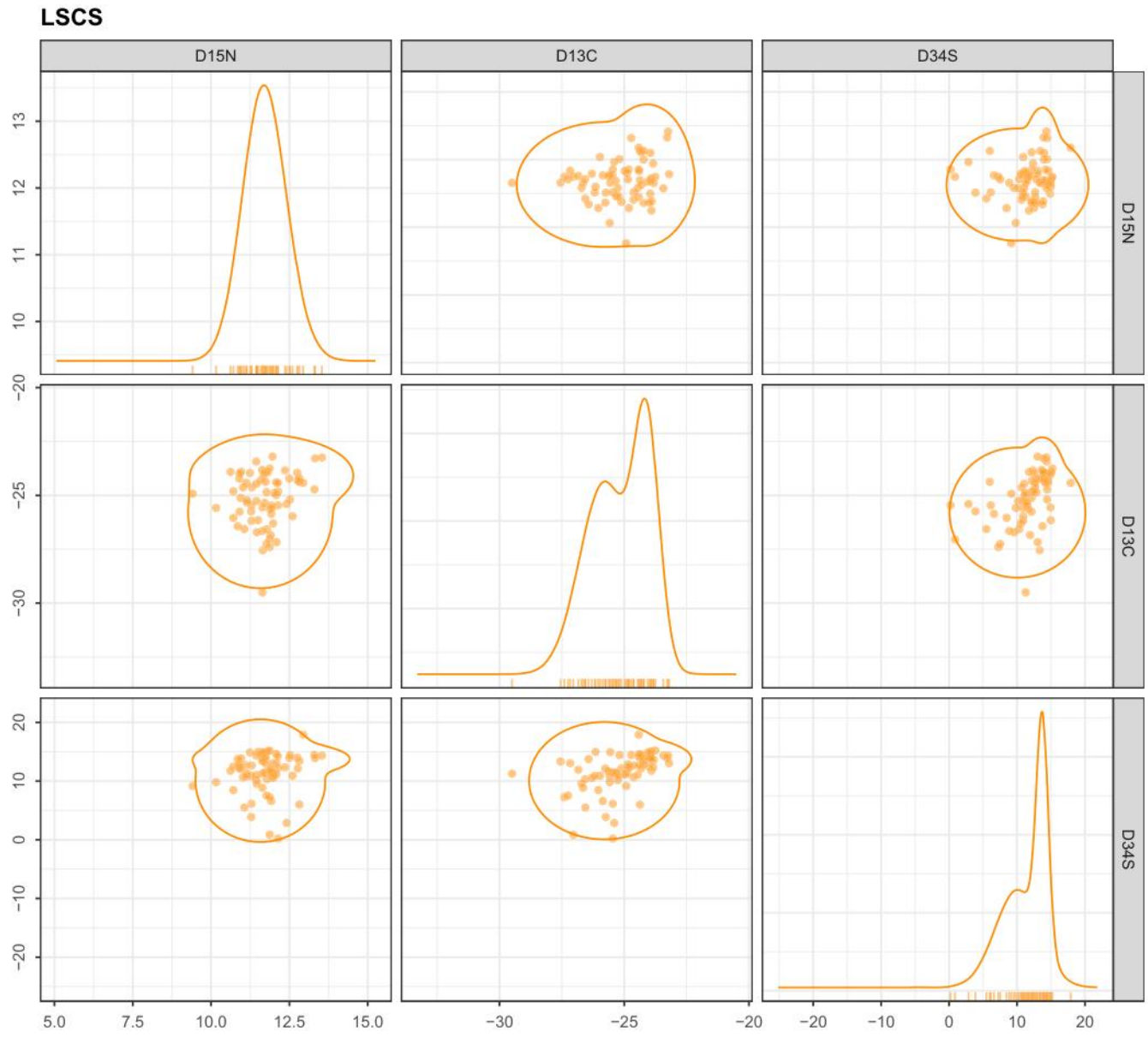

FIGURE B4 | Distribution of the three stable isotopes measured in the muscle tissue of *Coregonus sardinella* (LSCS). Contours represent the GMM hull estimated according to the proposed method, delineating the trophic niche region of this species in multivariate isotopic space.