


## RESEARCH ARTICLE

New approaches for delineating  $n$ -dimensional hypervolumesBenjamin Blonder<sup>1</sup>  | Cecina Babich Morrow<sup>2</sup> | Brian Maitner<sup>3</sup> | David J. Harris<sup>4</sup> | Christine Lamanna<sup>5</sup> | Cyrille Violle<sup>6</sup>  | Brian J. Enquist<sup>3,7</sup> | Andrew J. Kerkhoff<sup>2</sup>

<sup>1</sup>Environmental Change Institute, School of Geography and the Environment, University of Oxford, Oxford, UK; <sup>2</sup>Department of Biology, Kenyon College, Gambier, OH, USA; <sup>3</sup>Department of Ecology and Evolutionary Biology, University of Arizona, Tucson, AZ, USA; <sup>4</sup>Department of Wildlife Ecology and Conservation, University of Florida, Gainesville, FL, USA; <sup>5</sup>World Agroforestry Centre, Nairobi, Kenya; <sup>6</sup>CNRS, CEF, UMR 5175, Paul Valéry University of Montpellier – EPHE, Montpellier Cedex 5, France and <sup>7</sup>The Santa Fe Institute, Santa Fe, NM, USA

## Correspondence

Benjamin Blonder

Email: bblonder@gmail.com

## Funding information

UK Natural Environment Research Council, Grant/Award Number: NE/M019160/1; US National Science Foundation, Grant/Award Number: DEB-1556651; Kenyon College Summer Science; National Science Foundation, Grant/Award Number: DEB-1457812 and Macrosystems-1065861; European Research Council (ERC), Grant/Award Number: ERC-StG-2014-639706-CONSTRAINTS; French Foundation for Research on Biodiversity

Handling Editor: Sean McMahon

## Abstract

1. Hutchinson's  $n$ -dimensional hypervolume concept underlies many applications in contemporary ecology and evolutionary biology. Estimating hypervolumes from sampled data has been an ongoing challenge due to conceptual and computational issues.
2. We present new algorithms for delineating the boundaries and probability density within  $n$ -dimensional hypervolumes. The methods produce smooth boundaries that can fit data either more loosely (Gaussian kernel density estimation) or more tightly (one-classification via support vector machine). Further, the algorithms can accept abundance-weighted data, and the resulting hypervolumes can be given a probabilistic interpretation and projected into geographic space.
3. We demonstrate the properties of these methods on a large dataset that characterises the functional traits and geographic distribution of thousands of plants. The methods are available in version  $\geq 2.0.7$  of the `HYPERVOLUME` R package.
4. These new algorithms provide: (i) a more robust approach for delineating the shape and density of  $n$ -dimensional hypervolumes; (ii) more efficient performance on large and high-dimensional datasets; and (iii) improved measures of functional diversity and environmental niche breadth.

## KEYWORDS

functional diversity, functional space, hypervolume, kernel density estimation, niche, niche modelling, support vector machine

## 1 | INTRODUCTION

Over the past decade, numerous studies have used the  $n$ -dimensional hypervolume as a central concept for describing the functional diversity of communities (Barros, Thuiller, Georges, Boulangeat, & Münkemüller, 2016; Cornwell, Schwillk, & Ackerly, 2006; Díaz et al., 2016; Lamanna et al., 2014; Swenson & Weiser, 2014) and the niches of species and broader clades (Broennimann et al., 2007; Peterson, Soberon, & Pear, 2011; Soberón & Nakamura, 2009; Swanson et al., 2015; Tingley, Vallinoto, Sequeira, & Kearney,

2014). Originally proposed by Hutchinson (1957), this concept assumes that a system can be characterised by a set of independent axes, e.g. functional traits, resource requirements, or abiotic tolerances. These axes constitute an  $n$ -dimensional Euclidean space. A geometrical shape can then be delineated within this space and used to describe the size, position and geometry of the system. The shape may also be described as a probability distribution over these axes, with level sets corresponding to a range of possible geometries (Blonder, Lamanna, Violle, & Enquist, 2014; Carmona, de Bello, Mason, & Lepš, 2016a).

Despite the intuitive nature of the concept, determining how to delineate the shape of a hypervolume for a given dataset has proven to be difficult and controversial. First, there are multiple methods available to estimate a hypervolume, each with different underlying assumptions. For example, functional diversity has been estimated with dynamic range boxes (Junker, Kuppler, Bathke, Schreyer, & Trutznig, 2016), convex hulls (Villéger, Mason, & Mouillot, 2008), or multidimensional ellipses (Swanson et al., 2015). Second, niches can also be estimated with approaches such as generalised linear models, generalised additive models and range boxes (e.g. BIOCLIM) that also can be interpreted geometrically, especially in the case of presence-only data (Elith et al., 2006; Peterson et al., 2011). While there has been some comparisons of methods for both niche (Bahn & McGill, 2013; Blonder et al., 2014; Elith et al., 2006; Qiao, Escobar, Saupe, Ji, & Soberón, 2017) and trait data (Mason & de Bello, 2013; Schleuter, Daufresne, Massol, & Argillier, 2010), the methods used for trait data are rarely applied to niche data, or vice-versa. As a result, there is not necessarily a clear “best” way to delineate hypervolumes (Blonder, in review; Merow et al., 2014).

The choice of an appropriate method depends on the goals of the analysis and is further complicated by data limitations. For example, in the context of niche modelling, it has been argued that fundamental niches should have simple geometries (Blonder, Lamanna, Violle, & Enquist, 2017; Qiao et al., 2017). However, species interactions, dispersal, and variation in environmental space availability can result in realized niches with more complex shapes (Jackson & Overpeck, 2000; Soberón & Nakamura, 2009). Moreover, incomplete or biased sampling could yield falsely complex shapes for both niches and functional diversity applications. As a result, for some modelling applications and sampling regimes, complex shapes may not be preferred. In these cases, methods already exist to fit simple distributions and shapes to data (e.g. convex hulls, ellipses (Mouillot et al., 2014; Swanson et al., 2015)). However, there are fewer solutions for when more complex shapes are desired.

Here, we present a Monte Carlo approach to delineating hypervolumes that builds on the random sampling method of Blonder et al. (2014). The approach can describe complex shapes in high dimensionalities, measure the volume of these shapes, perform set operations on multiple shapes (e.g. intersections, similarity indices), predict suitability maps by projecting from hyperspace onto geographic space, and detect the presence of holes (Blonder, 2016a). The original Blonder et al. (2014) method has been the subject of much discussion in the literature. On the one hand, the method has become widely used for both functional diversity and realized niche modelling applications (e.g. Barros et al., 2016; Carvajal-Endara, Hendry, Emery, & Davies, 2017; Díaz et al., 2016; Lamanna et al., 2014; Loranger et al., 2016). On the other hand, there has also been debate concerning the edge delineation and probabilistic assumptions of the original algorithms (Blonder, 2016b; Carmona et al., 2016a; Carmona, de Bello, Mason, & Lepš, 2016b) and whether the method is useful for fundamental niche modelling where complex geometrical features may not be expected (Blonder et al., 2017; Qiao et al., 2017).

Here, we further develop hypervolume concepts and address current limitations. To do so requires addressing two general mathematical problems shared by all hypervolume delineation approaches (Blonder, in review). The first is to build a hypervolume function  $h(x)$  based on input data that maps an  $n$ -dimensional vector within a Euclidean space  $X$  to a scalar (presence vs. absence, or probability of occurrence). Example functions include a generalised linear model or a convex hull. The second problem is how to best evaluate  $h(x)$  over  $X$  to delineate the subset of points at which  $h(x)$  is above a certain value. This thresholding step is useful for delineating contour-based boundaries, and thus for enabling geometric interpretations of the hypervolume function.

In general, the hypervolume function  $h(x)$  may not have a parametric description, so the  $n$ -dimensional shape must be delineated by numerically evaluating the function. This evaluation is a significant computational challenge, because a naïve boundary delineation algorithm would need to evaluate  $h(x)$  at all points in the space, which is inefficient when  $X$  is a high-dimensional space. In order to avoid this inefficiency, algorithms must deal with the trade-off between ignoring irrelevant regions of the  $n$ -dimensional space, while still sampling it sufficiently to delineate the shape accurately.

In Blonder et al. (2014), we initially addressed the problem of defining  $h(x)$  using kernel density estimation (KDE). This method places a probability kernel function around each input data point to yield some amount of padding (a “bandwidth”), corresponding to unsampled but potentially sampleable regions of the space. The resulting kernel functions are added together across all the data points and normalised. This process yields an  $h(x)$  with high values close to the input data points and lower values far away from the input data points. In our original publication, we proposed using a multivariate uniform *hyperbox* kernel function, i.e. one with a constant probability density over a finite range, specified by the bandwidth.

We then addressed the problem of evaluating  $h(x)$  and defining the hypervolume using a sampling approach that delineates a uniformly random set of points within the hyperspace with known values of  $h(x)$ , all guaranteed to be within the hypervolume. This random sampling approach supports methods for approximating the geometry of  $h(x)$ . Because the random points are uniformly distributed, and have a known density, it is possible to calculate the volume of the shape by dividing the number of random points by the point density. Additionally, it is possible to obtain contour boundaries by removing all random points below a certain threshold value of  $h(x)$ . Approximate set operations (e.g. overlap measures) can be performed by determining when random points are sufficiently close to other random points to be considered part of the same shape. These operations can be carried out without needing to know the underlying analytical form of  $h(x)$ , enabling complex operations to be carried out on hypervolumes independent of the function(s) that generated them.

While the original hyperbox approach of Blonder et al. (2014) has proven to be useful, it has a number of limitations. Because the hyperbox kernel is flat, the probabilities do not decay smoothly towards zero at the boundaries of the shape. This leads to jagged “squared-off” hypervolume boundaries with step changes in probability density. Moreover, the

original algorithm did not weight input data. As a result, the approach is only semi-probabilistic (Blonder, 2016b; Carmona et al., 2016a,b) and the resulting hypervolume cannot be thresholded at an arbitrary quantile.

Here, we describe two new algorithms for delineating and evaluating the hypervolume function  $h(x)$ , implemented in the `HYPERVOLUME` R package, versions  $\geq 2.0.7$ . The new algorithms can weight input data and produce boundaries that are smoother and conform more closely to the input data. Furthermore, one of the new algorithms can produce continuously varying probability densities that decay smoothly towards the boundary of the hypervolume and can be thresholded at any desired quantile. We also highlight additional new functions in the package relevant to species distribution modelling and functional diversity analysis. We then illustrate and compare the performance and results of these new methods using data on plant functional traits and environmental tolerances. Code illustrating usage for all new functions is available within the R package help. Scripts to replicate the demonstration analyses are provided as online supporting information.

## 2 | NEW HYPERVOLUME CONSTRUCTION METHODS

We first present two new hypervolume construction methods. The first is a Gaussian kernel density estimation method that can be accessed via the `hypervolume_gaussian` function. Unlike the uniform, flat hyperbox kernel, the Gaussian kernel decays towards zero continuously in all directions. The second is a one-class support vector machine (SVM) estimation method that can be accessed via the `hypervolume_svm` function. A SVM provides a smooth fit around data that is insensitive to outliers, yields a boundary that classifies points as either “in” or “out” of the hypervolume, and is computationally viable in high-dimensional hyperspaces (Drake, Randin, & Guisan, 2006; Schölkopf, Williamson, Smola, Shawe-Taylor, & Platt, 1999). Properties of these algorithms are illustrated in Box 1.

The two new methods are strictly better than those we originally proposed and should replace the old hyperbox kernel method in all situations. If the investigator wants continuous probabilistic output, and believes the data are an unbiased sample from a probability distribution (i.e. they seek a “loose wrap” to the data), then the Gaussian KDE method should be used. This method is probably most appropriate for most functional diversity applications and for fundamental niche modelling applications where the properties of the distribution underlying the observed data are of interest. Alternatively, if the investigator believes that the extreme values in the sampled data represent the true boundaries of the data (i.e. they want a “tight wrap” to the data) and wants a sharp binary classification boundary, then the SVM method should be used, with parameters chosen to achieve the desired tightness of the fit to the data. This is probably most appropriate for most realized niche modelling applications where the limits of the observed data are of most interest. The SVM approach is also more appropriate in very high dimensionality analyses, because KDEs become under-constrained by data as dimensionality increases: there are too many

possible probability density functions that would be consistent with the observed data (Scott, 2015; Scott & Wand, 1991). As previously noted (Blonder, 2016a), we suggest that a KDE analysis conducted with  $m$  input data points should be conducted in at most  $n = \log m$  dimensions).

However, ultimately the choice of which method to use depends on the investigator's beliefs about the true structure of the data, which is fundamentally unknowable from any sample (Soberón & Nakamura, 2009). As with any tool, the quality of results depends on the investigator choosing when to use it appropriately.

We also have wrapped these new methods (and the original hyperbox method) within a generic `hypervolume` function that enables the investigator to specify the algorithm to be used (hyperbox kernel density estimation, Gaussian kernel density estimation, SVM). The functions provides default values for all parameters that should provide reasonable performance for datasets of  $\sim 2$ –8 dimensions and up to 10,000 data points. Parameters can be changed (as detailed below) to also provide good performance in more challenging cases.

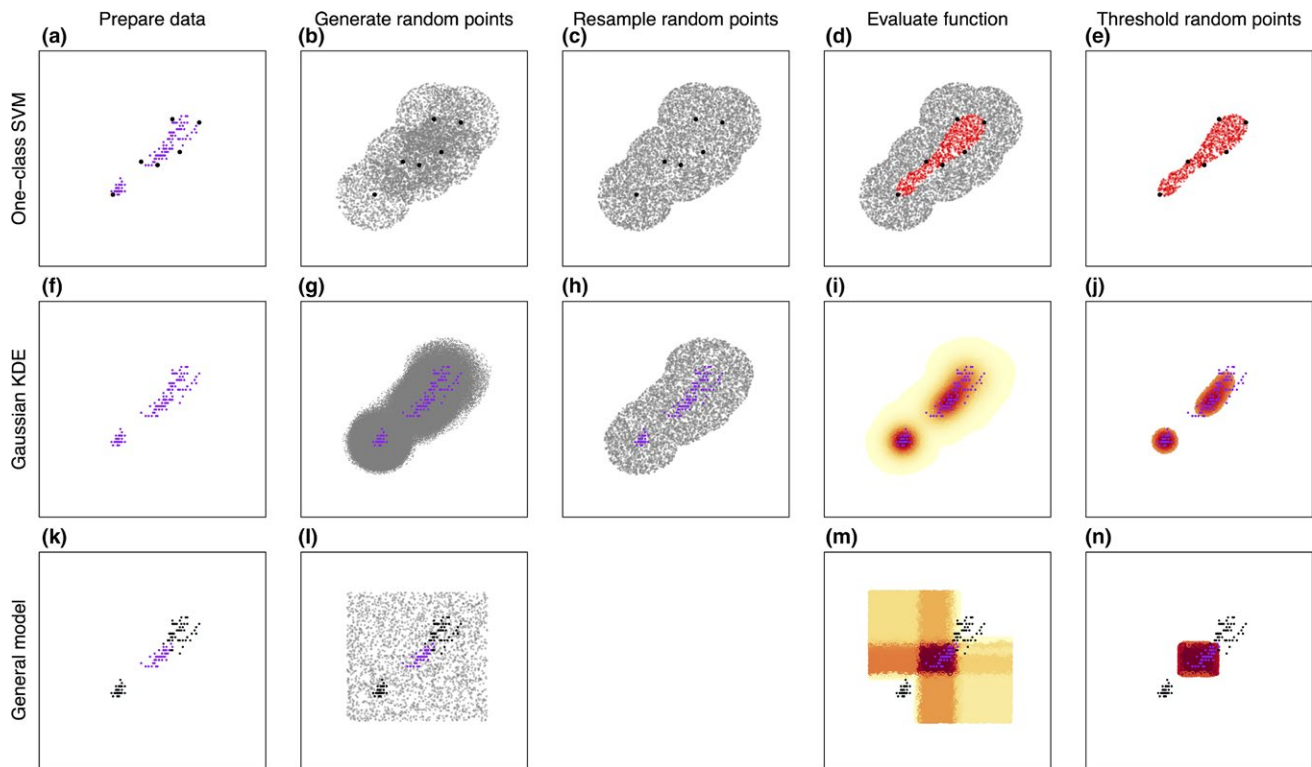
Regardless of the algorithm selected, the hypervolume functions all share similar data requirements. They are meant to be used on datasets with continuous, orthogonal axes, and it is helpful for interpretation of distances and volumes if axes have standardised units. Categorical data can potentially be ordinated with Gower transformation (Blonder, in review). Data ordination techniques (such as principal components analysis) can be used to improve axis orthogonality, and rescaling transformations (such as z-transforming) can help standardise distances (Blonder, in review; Carmona et al., 2016a). The generic `hypervolume` function also generates warnings for several of these data-related issues (e.g. highly correlated axes, insufficient number of data points, scale mismatches between axes, etc.) that we have previously highlighted (Blonder et al., 2014).

### 2.1 | Hyperelliptical uniform sampling

In order to define the set of possible points in  $X$  used to evaluate  $h(x)$  for both KDE and SVM methods, we use a hyperelliptical sampling algorithm. Elliptical sampling leverages the fact that points that fall inside the hypervolume are close to the data (Box 1a,f). Thus, for some definition of “close,” it is possible to sample only points that are “close” to the data and still obtain an accurate estimate of  $h(x)$ . This creates a trade-off: if the samples do not extend far enough away from the data, then portions of the hypervolume will be missed by the sampling procedure. On the other hand, sampling too far from the data is a waste of resources, since any such samples will fall outside of the hypervolume and be discarded without providing any useful information. Enclosing data within a fixed radius is also useful because hyperellipses are level sets of Gaussian functions, which are used for both the Gaussian KDE and SVM methods (see below). As such, sampling points within a hyperelliptical region guarantees that all of the subregions of  $X$  with values of  $h(x)$  above a certain threshold are sampled.

We therefore developed an algorithm for generating a uniformly random set of points from the union of the hyperellipses enclosing a

**Box 1 Demonstration of the hyperellipse random sampling method for delineating hypervolume boundaries for an input dataset (purple points) (compare to Box 1 in (Blonder et al., 2014) for hyperbox hypervolumes).**



(a) In a one-class support vector machine model, a subset of points are identified as support vectors (black points). (b) Uniformly random points are drawn from hyperellipses surrounding each support vector. (c) These points are resampled down to uniform density. (d) The support vector machine model is evaluated at each random point. (e) Positively classified points are retained to characterise the hypervolume.

(f) In a Gaussian kernel density estimate, all points contribute to the overall probability density. (g) Uniformly random points are drawn from hyperellipses surrounding each point. (h) These points are resampled to uniform density. (i) The kernel density estimate is calculated at each random point. (j) Points with values above a threshold enclosing a certain quantile of the probability or volume are retained to characterise the hypervolume.

(k) Using a function to construct a hypervolume from an arbitrary function (here, a random forest regression model), all points contribute to the overall estimate; here two-class input data are used (black, negatively classified points). (l) Uniformly random points are sampled from a hyperbox region. (m) The model is evaluated at each random point. (n) Points with values above a threshold are retained to characterise the hypervolume.

set of datapoints. We first sample uniformly random points from the unit hyperball, shifting and scaling to yield the location and covariance matrix of the desired hyperellipse. This procedure yields a set of points that are denser in regions of  $X$  that are covered by more than one hyperellipse (Box 1b,g). To reduce these random points to uniform density, we discard oversampled points in proportion to their density (using a recursive partitioning tree data structure to efficiently count the number of times each point is over-represented) (Bentley, 1975). The outcome of this process is the desired set of random points (Box 1c,h), uniformly distributed over a hypervolume that can be calculated as the volume of a single hyperellipse (determined analytically with standard formulas) multiplied by the number of input data points and by the mean inverse over-representation count.

## 2.2 | Delineating hypervolumes with a one-class support vector machine

We define a value of  $h(x)$  using a one-classification machine learning method for estimating the support of a probability distribution (Schölkopf, Platt, Shawe-Taylor, Smola, & Williamson, 2001) with the `svm` function from the `libsvm` R package (Meyer et al., 2012), which provides an interface to the `libsvm++` package (Chang & Lin, 2011). The SVM method uses positive observations of data (i.e. only one class of data) to identify regions of the hyperspace that should also be positively classified, yielding binary predictions (in vs. out, 1 vs. 0). We implemented the SVM with a radial basis function (RBF) kernel. Briefly, this means that  $h(x)$  is defined as a weighted sum of RBFs (each proportional to a multivariate Gaussian function) centred on the data

points. The weights in this weighted sum (denoted lowercase  $c_i$ ) are selected by an optimisation procedure that sets most weights to zero; the nonzero values are called “support vectors.” The same procedure automatically selects a threshold value (denoted  $\rho$ ): only the regions where  $h(x)$  exceeds this threshold are included in the hypervolume. The width of the RBF function is proportional to a user-selected tuning parameter,  $\gamma$  (`svm.gamma`), and the optimisation procedure is controlled by a second tuning parameter,  $\nu$ , (`svm.nu`). The value of  $\nu$  determines the upper and lower bound on the fraction of misclassification errors, and it is bounded between 0 and 1. Lower values of  $\nu$  yield lower in-sample prediction errors but potentially higher out-of-sample prediction errors, i.e. overfitting.

To efficiently sample points that do not have zero values of  $h(x)$ , we generate a hyperelliptical uniform sample of points that are close to the subset of data points that were identified as support vectors (i.e. the points that contribute to delineating the boundary of  $h(x)$ ): Box 1a,e. We define “close” by analytically solving for the maximum distance,  $d$ , away from a support vector that would still yield positive classification, assuming that all support vectors were in co-located (the case that would require the largest hyper-ellipse). This assumption yields a constraint equation that can be solved for  $d$ :  $e^{-\gamma d^2} \sum_i c_i - \rho = 0$ . This distance  $d$  is then scaled by the standard deviation of the data along each axis and used to determine the breadth of the sampled hyperellipses.

### 2.3 | Delineating hypervolumes with a Gaussian kernel density estimate

We define a value of  $h(x)$  as a mixture density, i.e. a sum of multivariate normal distributions with means centred at each data point and with diagonal covariance matrix scaled by the squared kernel bandwidth vector (Box 1i). Values of the bandwidth vector (`kde.bandwidth`) can either be chosen automatically using the `estimate_bandwidth` function, or specified by the investigator. As described previously (Blonder, 2016b), this function allows calculation of a Silverman bandwidth estimator (the default; optimal for axis-wise optimisation of normally distributed data), a plug-in estimator (Wand & Jones, 1994) and a cross-validation estimator (Duong & Hazelton, 2005). The first algorithm is computationally fast while the latter two algorithms are slower but have lower predictive error rates.

The Gaussian kernel density estimate in principle is non-zero everywhere in the hyperspace. To produce bounded output, we generate a set of uniformly random points using the hyperelliptical uniform sampling algorithm described above (Box 1f,h). The edge of this sample, i.e. the “closeness” of random points to the data, is determined by multiplying the bandwidth vector by a fixed number, `sd.count`. That is, the hyperellipses each enclose a region bounded by a certain number of standard deviations from the multivariate normal distribution centres. As `sd.count` becomes increasingly large, the probability density estimate becomes increasingly accurate. This initial boundary choice sets a maximum possible volume for the hypervolume and is necessary to ensure that the output does not have an infinite size. We next evaluate  $h(x)$  at each of these random points (Box 1i). We then finally retain points only with values of  $h(x)$  above a certain threshold (Box 1j). The

volume of the hypervolume is then the ratio of the number of retained random points to the number of random points that were tested. The default threshold encloses 95% of the probability density of the kernel density (via the biased estimate over  $X$ ). However, it is also possible to choose a threshold that encloses a different quantile (`quantile.requested`) of either the probability density or of the total enclosed volume (`quantile.requested.type`). The thresholding approach is fully described in the next section. Note that to generate smaller hypervolumes with less padding around the data, it is not recommended to decrease `sd.count`. While varying this parameter appears to vary the amount of “padding” around each data point, it does so only by truncating the estimate of the probability density and increasing bias in the overall result. Rather, the investigator should decrease `kde.bandwidth` to manipulate the level of padding around the input data. The value of `sd.count` should generally not be modified, and should never be decreased from its default value of 3. This reduces bias in hypervolume estimates but has higher computational costs due to the larger number of evaluations of  $h(x)$ .

### 2.4 | Parameter choice

All of the hypervolume construction methods also depend on a computational parameter: for `hypervolume_box`, `hypervolume_gaussian`, and `hypervolume_svm`, the parameter is called `samples.per.point`. For `hypervolume_general_model` and `expectation_convex`, it is called `num.samples`. This parameter determines the number of uniform random points used to represent values of  $h(x)$ . The default value of this parameter is chosen via a heuristic approach to increase with the square root of the dimensionality of the analysis, in order to provide more robust estimates of higher-dimensional shapes that may have complex forms. For the first three functions the value is chosen on a per-point basis, so that hypervolumes with different number of data points have the same overall sampling effort; the latter two functions can be run without specifying a dataset, so no such normalisation is performed. The investigator can further increase this parameter if desired. Larger values of this parameter produce more robust results at higher computational cost.

#### 2.4.1 | Quantile thresholds

To allow a more fully probabilistic description of hypervolumes, we also implement a general method (`hypervolume_threshold`) to choose one or more thresholds to delineate boundaries of a hypervolume with a given probability distribution. Many non-probabilistic functions can be given a probabilistic interpretation: Any  $h(x)$  that has a bounded integral can be normalised by this value and then treated as a probability density function. For example, a SVM-based hypervolume can be reconsidered as a uniform probability density over its extent.

The thresholding method generates a large range of uniformly spaced possible threshold values varying from the minimum to the maximum value in the hypervolume, then retains points with  $h(x)$  exceeding this threshold. At each threshold, the algorithm then calculates both the volume enclosed (proportional to the number of random



points) as well as the total probability density enclosed (proportional to the sum of the values of the random points). These values can be used to estimate an empirical cumulative distribution function and thus quantiles of the volume or probability density. The algorithm can then return either all of these nested hypervolumes, or a single hypervolume corresponding to a desired quantile value of the desired type (e.g. 95% of the total enclosed volume relative to the initial hypervolume) (Box 2). The nested hypervolumes can either be returned with their values intact, or “flattened” to uniform probability values throughout, i.e. as representing geometrical shapes. If the exact quantile value requested cannot be obtained, the next closest value is used instead.

The choice of threshold can be determined in several ways depending on the goal of analysis. For SVMs, there is only one possible threshold value, so no choice need be made. For KDEs, using a fixed algorithm for threshold choice (e.g. 95% quantile) is appropriate in most cases. In some applications (e.g. species distribution modelling), the threshold can instead be chosen to maximise some classification statistic, e.g. sensitivity or specificity. In other cases, the threshold can be varied to determine how the outcome of the analysis of interest (e.g. volume, overlap) changes. Analysing results as a function of multiple threshold values therefore provides a key approach to use the full probabilistic structure of the hypervolume. For example, holes in a hypervolume may be absent at low thresholds, but appear at higher thresholds where valleys in probability distributions become apparent (Blonder, 2016a).

### 3 | OTHER NEW FUNCTIONALITY

Since its initial publication, we have also implemented a range of other features and minor improvements in the R package.

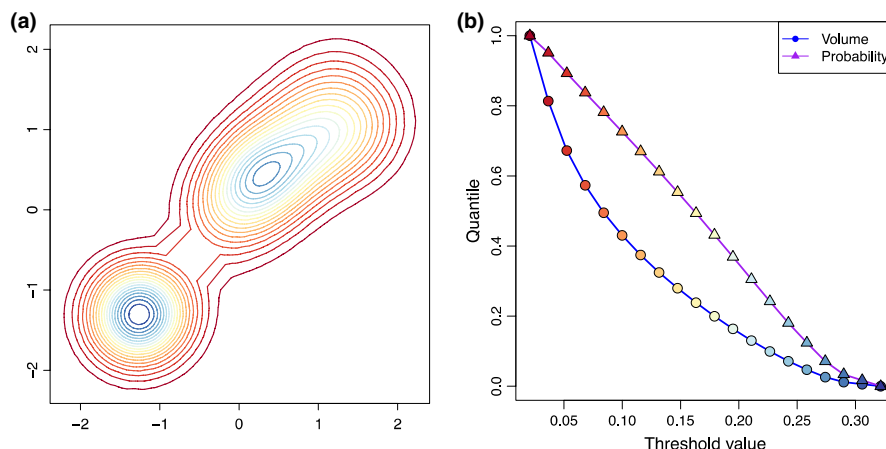
First, we introduce a function, `hypervolume_general_model`, that allows estimation of hypervolumes for any arbitrary function  $h(x)$ .

This new function can effectively map data from an  $n$ -dimensional Euclidean space into a one-dimensional Euclidean space with non-negative values. This approach may be useful for treating the outputs of, e.g. generalised linear models, generalised additive models, or random forests, as hypervolume functions. The investigator specifies a `model` object for which the generic `predict` R function can be called, as well as a `range.box` parameter, defining a set of minimum and maximum values along each hypervolume axis, and a `min.value` parameter. The function then samples a set of uniform random points from the hyperbox defined by these values, and evaluates the model object at each point, delineating values of  $h(x)$ . The function then retains only points where values of  $h(x)$  exceed `min.value`, and calculates the resulting volume as the fraction of retained points multiplied by the volume of the hyperbox. For models with binary output, setting `min.value` to zero is sufficient to delineate the hypervolume boundaries. For models with continuous output, the resulting hypervolume can be further thresholded at different quantile values, as described below.

The `hypervolume_general_model` function provides a flexible approach to identifying the shape of arbitrary functions using hypervolume concepts. It does have high computational cost in high dimensions, because the hyperbox sampling evaluates the entire Euclidean space, without the efficiency gains of the hyperellipse sampling used for the kernel density estimate and SVM models. The hyperbox approach requires the investigator to specify the region over which the model will be evaluated. This may lead to clipping of hypervolume boundaries but is also useful in constraining output from models that do not predict  $h(x) = 0$  at  $x = \pm\infty$ , e.g. some linear and generalised linear models.

The `HYPERVOLUME` package already includes an `expectation_convex` function that provides an approach for defining  $h(x)$  as the minimum convex polytope enclosing the data points. In the new package version, this function's performance has been dramatically

#### Box 2 Demonstration of the quantile thresholding algorithm



(A) Hypervolume boundaries can be delineated at a range of threshold values corresponding to different values of the underlying estimation function (colored regions). (B) Each of these regions encloses a different quantile fraction of the total volume or probability density. Thus, the investigator can specify a desired quantile fraction and the algorithm can then determine the appropriate threshold to use.

improved by the use of an adaptive hit-and-run sampling algorithm detailed in (Tervonen, van Valkenhoef, Baştürk, & Postmus, 2013).

We also include an approach to allow for weighting of data and variation/uncertainty within data. This approach can be used for example to account for abundance of species and/or intraspecific trait variation in community-level functional diversity analyses (Enquist et al., 2015). The `weight_data` function can be applied before running any of the hypervolume algorithms. Data point importance can be varied by duplicating each point a certain number of times before analysis, effectively adding copies of each observation into the data. Uncertainty and/or variation around each observation can also be incorporated within this function by replacing observations by a sample from a Gaussian distribution with a chosen standard deviation.

We provide a function `hypervolume_estimate_probability` that can generate geographic maps for species distribution modeling (e.g. Figure 6). This function equates suitability scores to estimated hypervolume function values at each point in hyperspace (Peterson et al., 2011). It estimates the probability density at arbitrary points in the hyperspace by taking a distance-weighted average of the values of  $h(x)$  at the random points. The distances are then raised to an arbitrary negative exponent `weight.exponent` (default,  $-1$ ) to give more or less weight to nearby points. This feature effectively assumes that the value of the test point is more likely to be equal to the value of  $h(x)$  at nearby test points. This function also underlies a new functional redundancy or uniqueness metric, `hypervolume_redundancy` (cf. Violle et al., 2017; Mouillot et al., 2014) consistent with some other usages of this term (Blonder, 2016b; Carmona et al., 2016a). The metric is defined as the value of  $h^2(x)$  evaluated at a test point  $x$ , which weights the value of  $h(x)$  by the probability of observing that value. This function can test whether given points are highly unique from others in the hypervolume.

We also now provide several summary statistics for hypervolume geometry. These include a function `hypervolume_overlap_statistics`, that implements indices describing the pairwise overlap between hypervolumes (e.g. Jaccard similarity, asymmetric unique fraction), and a function `hypervolume_distance` which determines the distance from a test point to either the closest point on the surface of the hypervolume or the centroid of the hypervolume. The centroid can be accessed directly via `get_centroid` and the volume via `get_volume`. These functions provide a convenient way to describe geometrical relationships between hypervolumes.

There are also several other minor changes to the package, including improvements in graphics quality for plotting, textual summary output, and manipulation of multiple hypervolumes using R's square bracket operators. Notably, the package now includes a `hypervolume_save_animated_gif` function that provides a convenient way to save rotating animations of three-dimensional projections of hypervolume (cf. Movie S1 in Lamanna et al., 2014).

Performance and usability is improved throughout. All major algorithms now process large datasets in smaller chunks to reduce memory usage (via a `chunk.size` argument), and provide status bars and diagnostics to better assess progress (via a `verbose` argument).

Default parameters have also been changed in many functions, yielding more robust results for many realistic use cases.

### 3.1 | Syntax and usage

A description of the R code used to access these new functions can be found in Table 1. Because of these extensive changes, the behaviour of the new package may differ from that of previous versions ( $<2.0.0$ ), but will be better. Syntax is very similar to previous versions, but not identical. As such, porting code to the new version is simple and requires only small syntax changes to core function calls. All package demos have been updated to reflect this new syntax as well. A detailed guide to recommended syntax changes is provided in Table S1. Our suggestions on R package usage for several common situations is provided in Table 2.

## 4 | DEMONSTRATION ANALYSES

We performed a pair of analyses demonstrating the use of these new functions on both functional trait and environmental niche data (with R code available in the online supporting information). These represent two of the most common applications of hypervolume methods. In both cases, data were drawn from the Botanical Information and Ecology Network (BIEN) database, which includes trait data and range maps for plant species in North and South America (<http://www.biendata.org>). Data were obtained using the BIEN R package.

First, we compiled plant functional trait data to create functional hypervolumes for each biome designated by the World Wildlife Fund (WWF) (Olson et al., 2001). This analysis can be replicated using code in Supporting Text S1 and S2. Second, we analysed bioclimatic data to create realized niche hypervolumes for a randomly selected set of 100 species of North and South American plants. The two separate analyses also apply the new methods to situations in which the density of input data is both high (bioclimatic data) and relatively sparse (functional trait data), relative to the observed range of data values.

### 4.1 | Functional trait analyses

For functional trait analyses, we selected height, seed mass, and specific leaf area as the axes for the hypervolumes. This suite of traits represent major axes in the plant economic spectrum that describe key aspects of plant ecology strategy, including physiology and life history (Westoby, 1998). The BIEN3 database included 1,544 plant species with coverage for all three traits.

In order to create the hypervolumes for all of the methods, the three traits for both taxa were log-transformed and then scaled. Hypervolumes were calculated for each of the 14 biomes by overlapping the BIEN3 range polygons with those of the terrestrial ecoregions. If a species' range overlapped with a biome, it was counted as present in that biome. This is a highly inclusive method for determining which

**TABLE 1** New functions and parameters in the `HYPERVOLUME` R package. Each function may also have additional arguments that are detailed in the R package documentation

Function	Description	Key arguments
<code>hypervolume</code>	Generic function to access all hypervolume methods. Checks data for common errors and warnings before proceeding	
<code>hypervolume_box</code>	Hypervolume estimation by hyperbox kernel density estimation, replicating functionality in earlier versions of the package	<code>kde.bandwidth</code> (bandwidth vector), <code>samples.per.point</code> (computational parameter)
<code>hypervolume_distance</code>	Calculates distance from test point to hypervolume	<code>type</code> (whether distance is to boundary or to centroid)
<code>hypervolume_estimate_probability</code>	Estimates probability density at a given point in hyperspace	<code>weight.exponent</code> (distance weighting power)
<code>hypervolume_gaussian</code>	Hypervolume estimation by Gaussian kernel density estimation	<code>kde.bandwidth</code> (bandwidth vector), <code>samples.per.point</code> (computational parameter), <code>sd.count</code> (computational parameter)
<code>hypervolume_general_model</code>	Estimates a hypervolume for an arbitrary model over a hyperbox region	<code>model</code> (arbitrary statistical model), <code>range.box</code> (hyperbox sampling region), <code>num.samples</code> (computational parameter), <code>min.value</code> (threshold value at which points are discarded)
<code>hypervolume_overlap_statistics</code>	Calculates a range of overlap statistics for two hypervolumes including Sørensen and Jaccard similarity	
<code>hypervolume_project</code>	Creates a geographic suitability map based on a set of input rasters and the probability density function within a hypervolume object	<code>rasters</code> (georeferenced layers to be used as predictors), <code>type</code> (binary vs. continuous output)
<code>hypervolume_redundancy</code>	Estimates functional redundancy metric at a given point in hyperspace as the squared probability density at that point	
<code>hypervolume_save_animated_gif</code>	Saves an animated GIF of a three-dimensional projection of a hypervolume	
<code>hypervolume_svm</code>	Hypervolume estimation by one-class support vector machine	<code>svm.nu</code> (error rate parameter), <code>svm.gamma</code> (smoothing parameter), <code>samples.per.point</code> (computational parameter), <code>sd.count</code> (computational parameter)
<code>hypervolume_threshold</code>	Calculates single or multiple flattened and thresholded hypervolumes at a specified threshold, volume quantile, or probability quantile	<code>quantile.requested</code> (quantile parameter), <code>quantile.requested.type</code> (type of quantile)
<code>weight_data</code>	Weights data for hypervolume analysis by abundance and optionally accounts for uncertainty in each observation	<code>weights</code> (vector of weighting values)

species are present in a biome, so the number of species in each biome is most likely overestimated, and many species occur in multiple biomes.

## 4.2 | Niche analyses

We randomly selected 100 plant species from the set of all North and South American species with at least 10 valid occurrence records in the BIEN3 database. Within each species' occurrence records, climate data were compiled from Worldclim at 10 km resolution (Hijmans, Cameron, Parra, Jones, & Jarvis, 2005). We selected three climate variables for our analyses: mean annual temperature, mean annual precipitation, and precipitation in warmest quarter/(precipitation in warmest quarter + precipitation in coldest quarter). Climate layers were z-transformed (centred relative to mean and scaled relative to standard deviation) prior to hypervolume construction.

## 4.3 | Parameters

For each of the new algorithms (Gaussian and SVM), we examined the sensitivity of the delineated hypervolumes to variation in the underlying parameters.

For the Gaussian KDE method, we varied the bandwidth parameter to determine its effects on the hypervolumes created. We started with the default Silverman bandwidth estimator since it is the simplest and least computationally intensive, and thus most likely the method selected by researchers. For all trait hypervolume analyses, we used a single bandwidth value corresponding to the overall Silverman bandwidth for the species across all 14 biomes. This baseline bandwidth was then used as the bandwidth for each individual biome. For the niche hypervolume analyses, the bandwidth was estimated separately for each species using the same algorithm. In both analyses, we then



**TABLE 2** Guidance for common usage situations

Situation	Guideline
Data have unordered categorical axes	Do not use hypervolume algorithms
Data have ordered categorical axes	Do not use hypervolume algorithms. With caution, convert data to continuous axes via, e.g. Gower transformation
Data have different units or scales	Rescale data before analysis
Data have missing observations	Reduce data to only complete cases and/or reduce dimensionality
Data have few observations	For KDE, reduce dimensionality of analysis until $\log_e m > n$
Data dimensionality is high	Use SVM instead of KDE
Data axes are highly correlated	Perform dimensionality reduction, e.g. via principal components analysis
Don't know what threshold to choose (for KDE)	Use quantile-based or classification-statistic-based algorithm to auto-select threshold, or repeat analyses for range of thresholds to determine sensitivity of result to threshold choice
Don't know what bandwidth to choose (for KDE) when comparing datasets	Use the same algorithm (e.g. Silverman or plug-in estimator) to choose a bandwidth for each dataset, or use a fixed bandwidth value across datasets
Trying to compare hypervolumes for datasets with very different number of observations (for KDE)	Resample data to fixed number of observations, or use algorithm to auto-select threshold and bandwidth (see above)
Results are numerically unstable or unrealistic	Increase values of computational parameters
R functions take too long to run	Be patient. Increase memory allocation to R. Reduce dimensionality of the analysis. With <i>extreme</i> caution, reduce values of computational parameters

varied bandwidth by multiplying the baseline bandwidth by a factor of 0.75 and a factor of 1.5 to demonstrate a range of parameter values and hypervolume sizes.

For the SVM method, we varied the two parameters,  $\nu$  and  $\gamma$ , starting with the default values  $\nu = 0.01$  and  $\gamma = 0.5$ . We allowed the parameter  $\nu$  to assume the values 0.01, 0.1 and 0.5 and  $\gamma$  the values 0.1, 0.5 and 2.5. We then created SVM hypervolumes using all nine possible combinations of these values to show the respective and combined effects of both parameters along a range of values.

#### 4.4 | Comparisons

We compared the total volume of the hypervolume delineated by each method relative to that delineated by the original hyperbox method. These comparisons were made for both trait and niche hypervolumes. We also made more qualitative comparisons by examining hypervolume shape for each method across the range of chosen parameter values. To examine differences in set operations, we measured the pairwise overlap of the trait hypervolumes for three of the biomes: temperate broadleaf and mixed forests; boreal forests/taiga; and temperate grasslands, savannas and shrublands. The fractional overlap between the hypervolumes was calculated by the intersection volume divided by the union volume (Jaccard similarity). The three biomes were selected based on their relatively good species coverage and varied degrees of functional trait overlap.

#### 4.5 | Geographic analyses

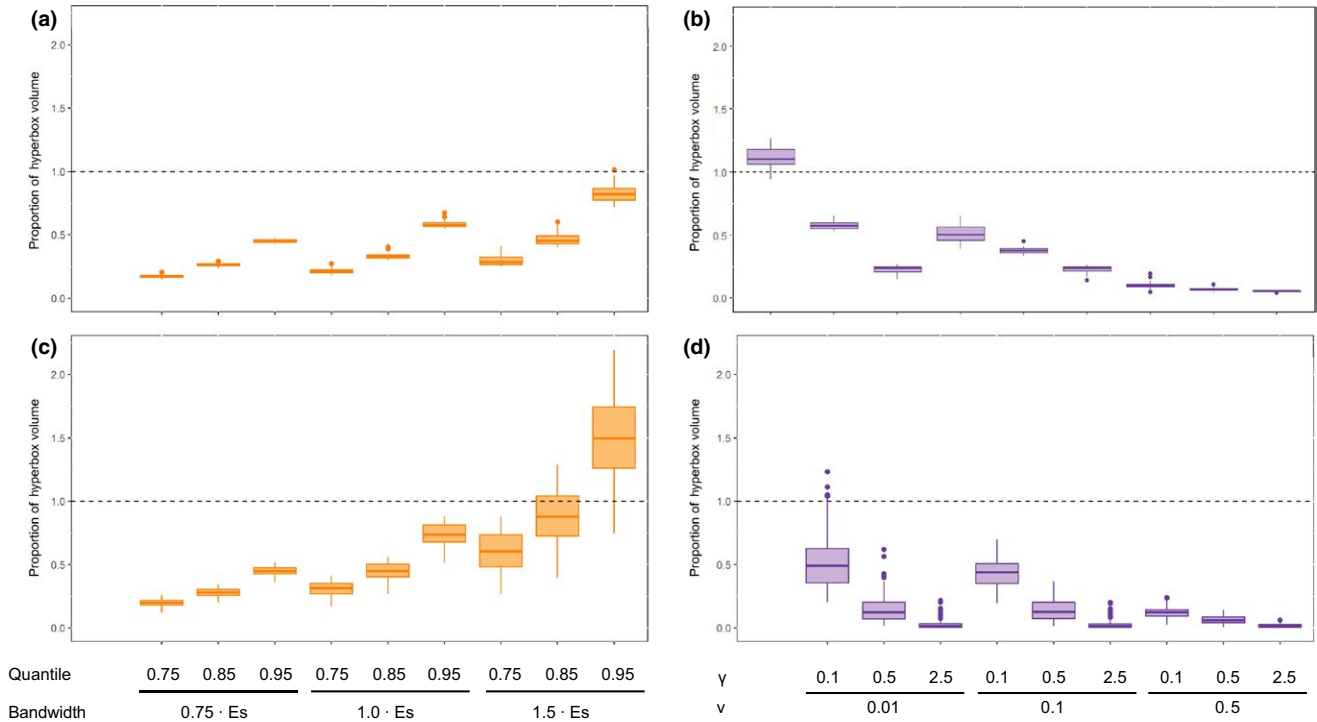
To demonstrate species distribution modeling functionality, we selected one species for which presence data were available, *Allium*

*parvum* (Alliaceae). We used the `hypervolume_project` function to map this species' potential distribution in North America using the Worldclim climate raster data described above. We generated distribution maps using both the KDE and SVM methods. We also produced a comparison map (Figure 6) with the same presence data using a standard maximum entropy method (Phillips, Anderson, & Schapire, 2006) using the default settings in the `dismo` R package.

## 5 | RESULTS

### 5.1 | Hypervolume size

The various hypervolume methods and parameter combinations give rise to a wide range of values for the volume. The overall volume depends on both the method and parameter values chosen, consistent with the different assumptions that underlie each method. There are also differences in volumes between the two datasets. For the functional trait analyses, the Gaussian hypervolumes are smaller than the hyperbox hypervolume, ranging from a fifth of the hyperbox volume to almost equal values (Figure 1a). The climate niche Gaussian hypervolumes have relative volumes similar to those of their functional trait counterparts for bandwidth factors of 0.75 and 1. For Gaussian hypervolumes with a bandwidth factor of 1.5, however, the climate niche volumes at probability quantiles 0.85 and 0.95 reach and exceed the hyperbox volume (Figure 1c). Across both datasets, however, the SVM hypervolumes are generally smaller than the corresponding hyperbox volume, with the exception of the SVM hypervolume with  $\nu = 0.01$  and  $\gamma = 0.1$ , the lowest parameter values tested (Figure 1b,d).



**FIGURE 1** Ratio of the hypervolume volume to hyperbox volume for functional trait hypervolumes for 14 biomes (a and b) and the niche hypervolumes of 100 plant species (c and d) for each method and parameter combination. (a) Gaussian and (b) SVM hypervolumes of functional trait data. (c) Gaussian and (d) SVM hypervolumes of niche data. Gaussian hypervolumes are grouped by threshold value and bandwidth as a factor of the Silverman estimate ( $E_s$ ). SVM hypervolumes are grouped by parameters  $\gamma$  and  $\nu$

Using default parameters, the relative ordering of volumes was fairly similar for both datasets. The hyperbox hypervolumes are consistently larger than the default Gaussian (Silverman bandwidth estimator, probability quantile = 0.95) and SVM hypervolumes ( $\nu = 0.1$ ;  $\gamma = 0.5$ ) across both the functional trait and climate niche analyses. For the functional traits analysis, the default Gaussian hypervolumes were very similar in size to the default SVM hypervolumes (Figure 1a,b). In the climate niche analysis, the default Gaussian hypervolumes were slightly smaller than the hyperbox, followed by the default SVM hypervolume, which was much smaller than the other two methods. Thus, the dataset being analysed has a moderate impact on the volume at default parameter values, but in both datasets, the new methods at default parameters produce smaller hypervolumes than the original hyperbox method.

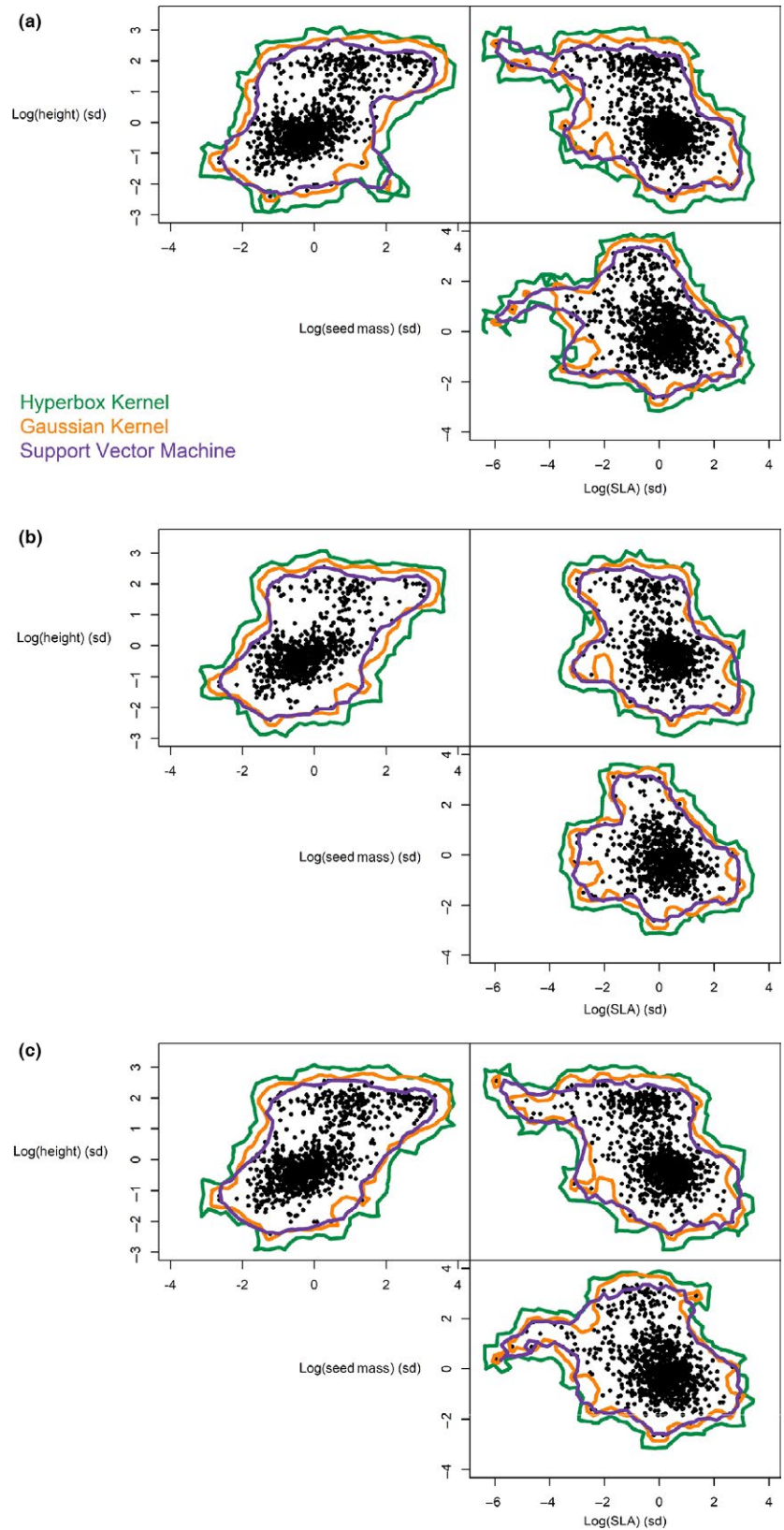
In the SVM hypervolumes, the parameters  $\nu$  and  $\gamma$  affect hypervolume size similarly in both analyses. As  $\gamma$  increases, volume decreases, and as  $\nu$  increases, volume decreases. At higher values of both parameters, however, the effects of increasing the other become less pronounced (Figure 1b,d).

In the Gaussian hypervolumes, the bandwidth and quantile choice have the same effects on  $\alpha$  hypervolume size in both analyses. As bandwidth increases, volume increases, and as quantile increases, volume increases. These relationships were apparent in both datasets, although the climate niche hypervolumes consistently had larger volumes relative to the hyperbox than their functional trait counterparts (Figure 1c,d).

## 5.2 | Hypervolume shape

Choice of method impacts the shape of the hypervolume (Figure 2). The hyperbox hypervolumes are generally the most jagged and blocky, while the Gaussian and SVM hypervolumes are smoother. The hyperbox hypervolumes are most influenced by extreme points, creating projections from the main body of the shape. The SVM hypervolumes display more concavities than either the hyperbox or Gaussian hypervolumes. The Gaussian hypervolumes are often the smoothest and most regular, although they often include “islands” in the hyperspace around more extreme data points.

Varying the parameters for the different methods affects the resulting shape of the hypervolume. For SVM hypervolumes, the size of the hypervolume decreases as both  $\nu$  and  $\gamma$  increase. Varying these two parameters decreases hypervolume size in different ways, however. As  $\nu$  increases while  $\gamma$  is held constant, the size of the hypervolume shrinks, but the shape remains fairly similar. Increasing  $\gamma$  while holding  $\nu$  constant also shrinks the hypervolume, but this change in size is driven by increased concavity or irregularity in the hypervolume's shape (Figure 3). For the kernel density estimation methods, changes in bandwidth leave the overall shape of the hypervolume constant while changing size. Increasing quantile threshold, however, modifies the shape by pulling the hypervolume's borders towards more extreme points (Figure 4). These shapes represent the location of the niche in the hyperspace, so changes in the shape resulting from method/parameter choice determine the placement of the niche.

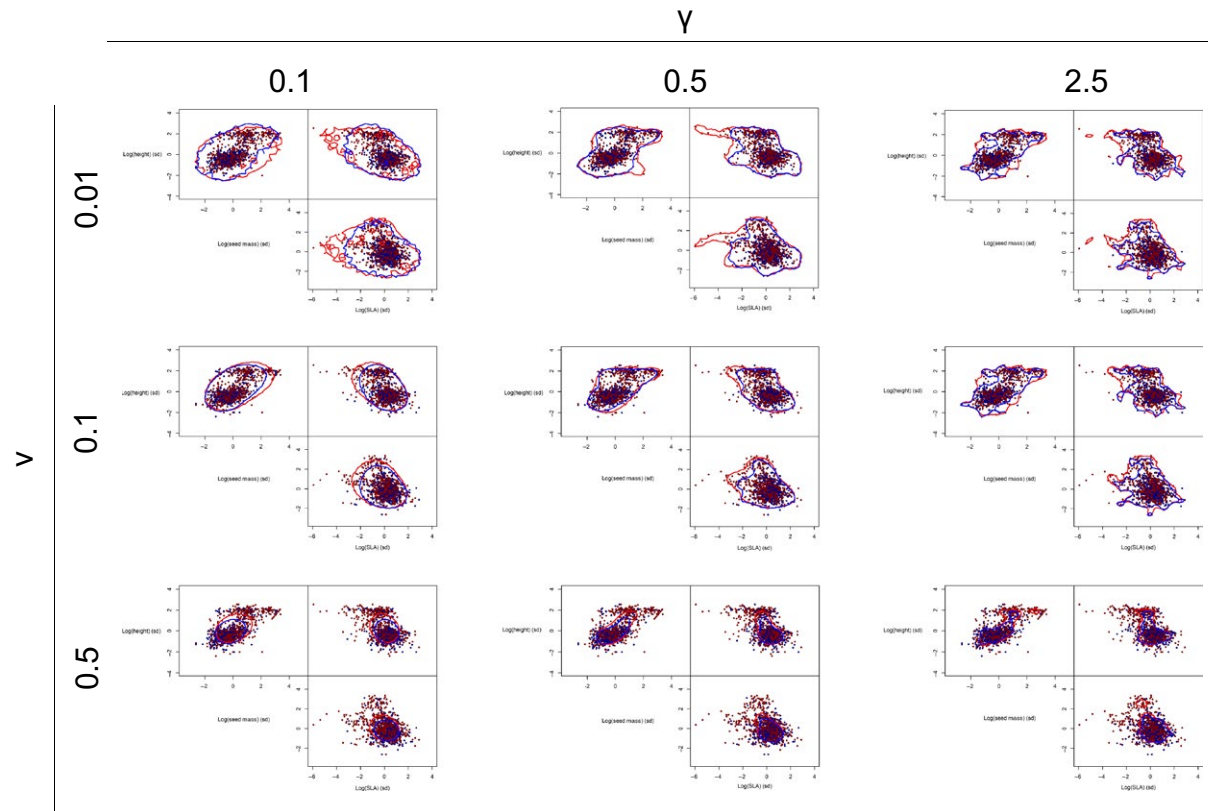


**FIGURE 2** Hyperbox, Gaussian, and support vector machine hypervolumes at default parameters for species in (a) temperate broadleaf and mixed forests, (b) boreal forests/taiga, and (c) temperate grasslands, savannas and shrublands

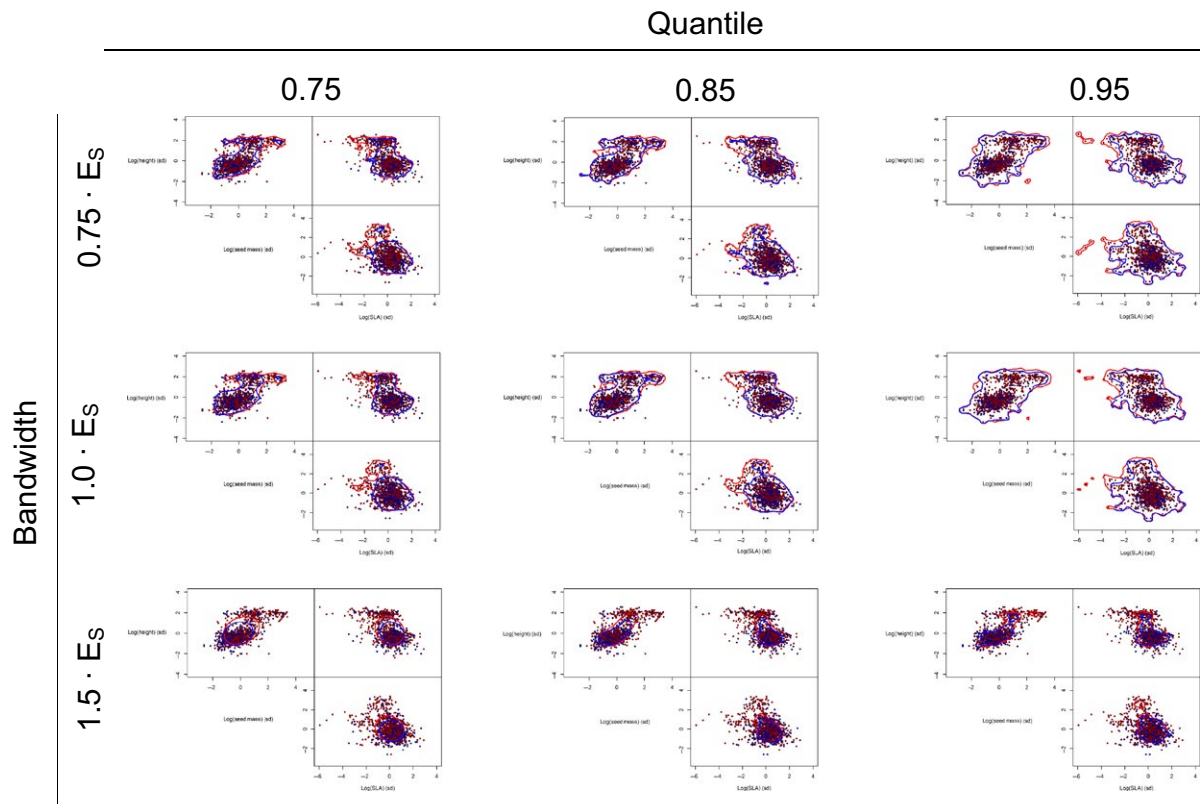
### 5.3 | Hypervolume overlap

Varying the hypervolume method and chosen parameter values does not drastically change the results of hypervolume set operations.

Regardless of method, the temperate forest and temperate grassland hypervolumes were much more similar to each other than either was to the boreal forest (Figure 5). This qualitative relationship was preserved across methods and parameter combinations. Thus, method



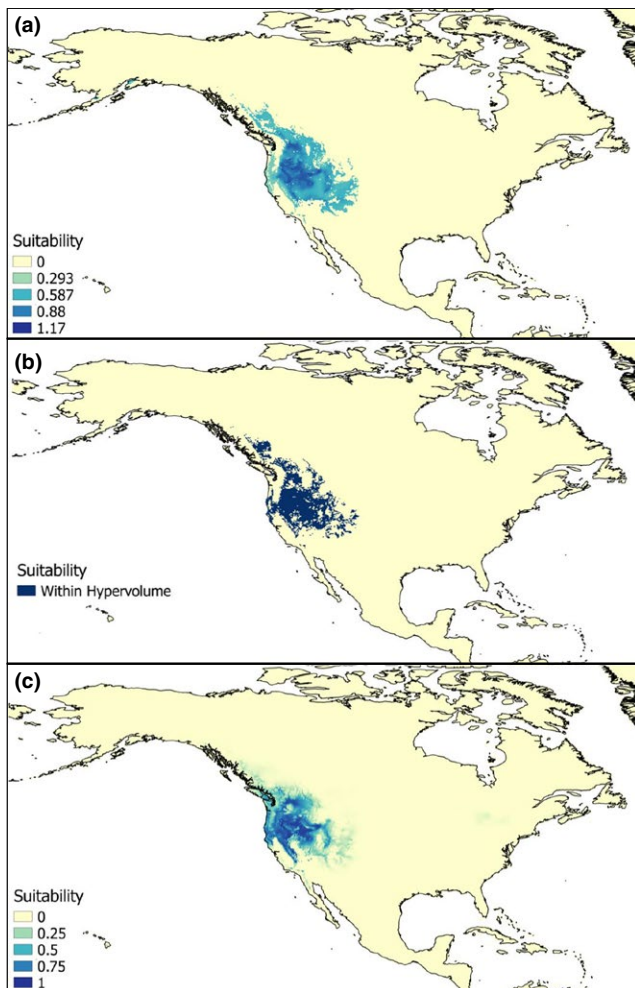
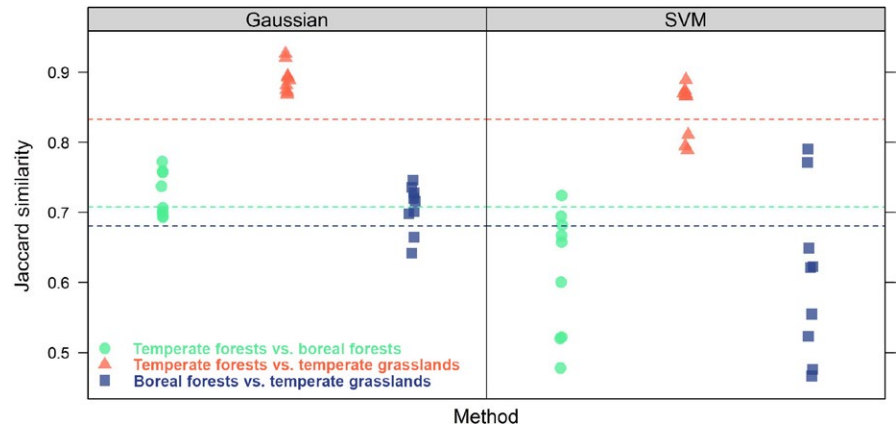
**FIGURE 3** Support vector machine hypervolumes of plants in temperate forests (red) and boreal forests (blue) for various combinations of parameter values



**FIGURE 4** Gaussian hypervolumes of plants in temperate forests (red) and boreal forests (blue) for varying values of bandwidth as a factor of the Silverman estimate ( $E_s$ ) and varying thresholds



**FIGURE 5** Pairwise fractional overlap (Jaccard similarity) of hypervolumes between temperate forests, boreal forests, and temperate grasslands. Fractional overlap was calculated by dividing the volume of the intersection of the two hypervolumes by the volume of their unions. Individual points correspond to different parameter combinations for the different methods. The dotted lines correspond to the fractional overlap of the hyperbox hypervolumes for the biome comparison indicated by color



**FIGURE 6** Species distribution model comparison for the wild onion, *Allium parvum*. (a) Continuous suitability values predicted from the Gaussian KDE method. (b) Binary predictions from SVM method. (c) Continuous suitability predictions from a standard maximum entropy method. Maps were generated using the same presence data and climate layers using default settings

choice should not greatly affect which niches are most similar to one another. On the other hand, the quantitative estimates for the Jaccard similarity vary both between and within methods (Figure 5).

Jaccard similarity values were slightly higher for the Gaussian hypervolumes than the SVM hypervolumes. The SVM hypervolumes had a wider range of similarity values, indicating that parameter choices had more of a varied impact on the overlap of SVM hypervolumes than of Gaussian hypervolumes. In conclusion, the qualitative relationships between measurements of overlap are robust to choices of method and parameter, although quantitative values may vary.

## 5.4 | Geographic distributions

Both the KDE and SVM methods produced visually similar potential geographic ranges for *A. parvum*. The KDE method yields continuous suitability scores, while the SVM method yields binary presence/absence predictions. The overall geographic maps overlapped in key areas, e.g. the Sierra Nevada and Great Basin desert. The KDE model predicted higher suitability in southwestern Colorado and southern British Columbia than the maximum entropy range models, while the SVM model was more congruent. These results demonstrate that these new methods provide comparable and reasonable outputs for correlative species distribution modelling applications.

## 6 | DISCUSSION

We have presented several novel methods for estimating hypervolumes that extend the approach we originally proposed (Blonder et al., 2014) and aim to address a range of issues around these ideas. Our new Gaussian KDE and SVM methods provide complementary approaches to loosely or closely wrap the data, and our new thresholding and weighting algorithms provide more robust ways to interpret hypervolumes in probabilistic contexts.

Our demonstration analyses showed that these new methods provide flexibility in the type of shape to be delineated. As expected, the shape and size of hypervolumes varied with method, with consistent and predictable effects of variation in parameters for each. Moreover, the relative ordering of sizes and overlaps was generally consistent across methods and across datasets,



suggesting that these methods all are capable of describing real biological variation.

The new methods do introduce a set of biologically relevant parameters into an analysis. These include the bandwidth vector for the Gaussian KDE and the  $v$  and  $\gamma$  parameters for the SVM. In the package we provide default values for each of these parameters that should yield reasonable performance, and which should enable robust comparison of data. However, results certainly vary depending on the values of these parameters. Ultimately, the choice of parameters should reflect investigator belief about how best to resolve the trade-off between false positive and false negative errors. Cross-validation approaches could be used to select these parameters according to some optimality criteria, but we believe this flexibility is a benefit rather than a detriment. Other models with no free parameters also make very strong assumptions about the types of errors that should be admitted. For example, a parameter-free range box model has very low false positive error rates, because all observed data are included within the model, but can have high false negative error rates, because any unmeasured data outside the range of the measured data will always be misclassified. This is not the case for the more flexible KDE and SVM methods we present here.

The growing interest in using  $n$ -dimensional hypervolumes to answer biological questions indicates a parallel need to consolidate operational concepts and develop robust estimation methods. The pluses and minuses inherent in any of these algorithms have been extensively discussed previously (Blonder, 2016a; Blonder et al., 2014; Qiao et al., 2017) and remain relevant here. We have not tried to present a comprehensive statistical comparison of these hypervolume methods to each other and to other existing methods (Elith et al., 2006; Junker et al., 2016; Qiao, Soberón, & Peterson, 2015; Swanson et al., 2015). A comprehensive comparative study would require exhaustive exploration of the different methods using a common set of simulated data with known statistical properties, subject to a variety of sampling scenarios. Such a set of simulations is beyond the scope of this work, and will be the subject of a forthcoming publication.

## ACKNOWLEDGEMENTS

B.B. was supported by a UK Natural Environment Research Council independent research fellowship (NE/M019160/1). A.J.K. and C.B.M. were supported by a collaborative research grant from the US National Science Foundation (DEB-1556651), and by the Kenyon College Summer Science programme. B.J.E. was supported by National Science Foundation award DEB-1457812 and Macrosystems-1065861. C.V. was supported by the European Research Council (ERC) Starting Grant Project “Ecophysiological and biophysical constraints on domestication of crop plants” (Grant ERC-StG-2014-639706-CONSTRAINTS) and by the French Foundation for Research on Biodiversity (FRB; [www.fondationbiodiversite.fr](http://www.fondationbiodiversite.fr)) in the context of the CESAB project “Causes and consequences of functional rarity from local to global scales” (FREE).

## AUTHORS' CONTRIBUTIONS

B.B. and C.B.M. made equal contributions. B.B. and D.H. developed hypervolume methods and software. C.B.M. and B.M. tested methods and performed analyses. C.B.M., B.M. and A.J.K. conceived analyses. All authors contributed to writing the manuscript.

## DATA ACCESSIBILITY

The `HYPERVOLUME` software package is available on the CRAN archive at <https://CRAN.R-project.org/package=hypervolume>.

Data required to execute Supporting Text 1–3 are available freely from the BIEN database (<http://bien3.org/>) and are automatically downloaded from within the Supporting Text scripts using the `BIEN` R package, available on CRAN at <https://cran.r-project.org/package=BIEN>.

## ORCID

Benjamin Blonder  <http://orcid.org/0000-0002-5061-2385>

Cyrille Violle  <http://orcid.org/0000-0002-2471-9226>

## REFERENCES

- Bahn, V., & McGill, B. J. (2013). Testing the predictive performance of distribution models. *Oikos*, 122, 321–331.
- Barros, C., Thuiller, W., Georges, D., Boulangeat, I., & Münkemüller, T. (2016).  $N$ -dimensional hypervolumes to study stability of complex ecosystems. *Ecology Letters*, 19, 729–742.
- Bentley, J. L. (1975). Multidimensional binary search trees used for associative searching. *Communications of the ACM*, 18, 509–517.
- Blonder, B. (2016a). Do hypervolumes have holes? *The American Naturalist*, 187, E93–E105.
- Blonder, B. (2016b). Pushing past boundaries for trait hypervolumes: A Response to Carmona et al. *Trends in Ecology & Evolution*, 31, 665–667.
- Blonder, B. (in review). Hypervolume concepts in niche- and trait-based ecology. *Ecography*, <https://doi.org/10.1111/ecog.03187>.
- Blonder, B., Lamanna, C., Violle, C., & Enquist, B. J. (2014). The  $n$ -dimensional hypervolume. *Global Ecology and Biogeography*, 23, 595–609.
- Blonder, B., Lamanna, C., Violle, C., & Enquist, B. (2017). Using  $n$ -dimensional hypervolumes for species distribution modeling: A response to Qiao et al. (2016). *Global Ecology and Biogeography*, 26, 1071–1075.
- Broennimann, O., Treier, U. A., Müller-Schärer, H., Thuiller, W., Peterson, A. T., & Guisan, A. (2007). Evidence of climatic niche shift during biological invasion. *Ecology Letters*, 10, 701–709.
- Carmona, C. P., de Bello, F., Mason, N., & Lepš, J. (2016a). Traits without borders: Integrating functional diversity across scales. *Trends in Ecology & Evolution*, 31, 382–394.
- Carmona, C. P., de Bello, F., Mason, N. W. H., & Lepš, J. (2016b). The density awakens: A reply to Blonder. *Trends in Ecology & Evolution*, 31, 667–669.
- Carvajal-Endara, S., Hendry, A. P., Emery, N. C., & Davies, T. J. (2017). Habitat filtering not dispersal limitation shapes oceanic island floras: Species assembly of the Galápagos archipelago. *Ecology Letters*, 20, 495–504.
- Chang, C.-C., & Lin, C.-J. (2011). `LIBSVM`: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2, 27.
- Cornwell, W. K., Schwillk, D. W., & Ackerly, D. D. (2006). A trait-based test for habitat filtering: Convex hull volume. *Ecology*, 87, 1465–1471.

- Díaz, S., Kattge, J., Cornelissen, J. H. C., Wright, I. J., Lavorel, S., Dray, S., ... Gorné, L. D. (2016). The global spectrum of plant form and function. *Nature*, 529, 167–171.
- Drake, J. M., Randin, C., & Guisan, A. (2006). Modelling ecological niches with support vector machines. *Journal of Applied Ecology*, 43, 424–432.
- Duong, T., & Hazelton, M. L. (2005). Cross-validation bandwidth matrices for multivariate kernel density estimation. *Scandinavian Journal of Statistics*, 32, 485–506.
- Elith, J., Graham, C. H., Anderson, R. P., Dudík, M., Ferrier, S., Guisan, A., ... Zimmermann, N. E. (2006). Novel methods improve prediction of species' distributions from occurrence data. *Ecography*, 29, 129–151.
- Enquist, B. J., Norberg, J., Bonser, S. P., Violle, C., Webb, C. T., Henderson, A., ... Savage, V. M. (2015). Scaling from traits to ecosystems. *Advances in Ecological Research*, 52, 249–318.
- Hijmans, R. J., Cameron, S. E., Parra, J. L., Jones, P. G., & Jarvis, A. (2005). Very high resolution interpolated climate surfaces for global land areas. *International Journal of Climatology*, 25, 1965–1978.
- Hutchinson, G. (1957). Concluding remarks. *Cold Spring Harbor Symposia on Quantitative Biology*, 22, 415–427.
- Jackson, S. T., & Overpeck, J. T. (2000). Responses of plant populations and communities to environmental changes of the late Quaternary. *Paleobiology*, 26, 194–220.
- Junker, R. R., Kuppler, J., Bathke, A. C., Schreyer, M. L., & Trutschig, W. (2016). Dynamic range boxes – A robust nonparametric approach to quantify size and overlap of  $n$ -dimensional hypervolumes. *Methods in Ecology and Evolution*, 7, 1503–1513.
- Lamanna, C., Blonder, B., Violle, C., Kraft, N. J. B., Sandel, B., Šimová, I., ... Enquist, B. J. (2014). Functional trait space and the latitudinal diversity gradient. *Proceedings of the National Academy of Sciences*, 111, 13745–13750.
- Loranger, J., Blonder, B., Garnier, É., Shipley, B., Vile, D., & Violle, C. (2016). Occupancy and overlap in trait space along a successional gradient in Mediterranean old fields. *American Journal of Botany*, 103, 1050–1060.
- Mason, N. W., & de Bello, F. (2013). Functional diversity: A tool for answering challenging ecological questions. *Journal of Vegetation Science*, 24, 777–780.
- Merow, C., Smith, M. J., Edwards, T. C., Guisan, A., McMahon, S. M., Normand, S., ... Elith, J. (2014). What do we gain from simplicity versus complexity in species distribution models? *Ecography*, 37, 1267–1281.
- Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., Leisch, F., Chang, C., & Lin, C. (2012). Misc functions of the Department of Statistics (e1071). TU Wien, Version, 1.6-4.
- Mouillot, D., Villéger, S., Parravicini, V., Kulbicki, M., Arias-González, J. E., Bender, M., ... Vigliola, L. (2014). Functional over-redundancy and high functional vulnerability in global fish faunas on tropical reefs. *Proceedings of the National Academy of Sciences*, 111, 13757–13762.
- Olson, D. M., Dinerstein, E., Wikramanayake, E. D., Burgess, N. D., Powell, G. V. N., Underwood, E. C., ... Morrison, J. C. (2001). Terrestrial Ecoregions of the World: A New Map of Life on Earth: A new global map of terrestrial ecoregions provides an innovative tool for conserving biodiversity. *BioScience*, 51, 933–938.
- Peterson, A. T., Soberón, J., & Pear, R. G. (2011). *Ecological niches and geographic distributions* (MPB-49). Princeton, NJ: Princeton University Press.
- Phillips, S. J., Anderson, R. P., & Schapire, R. E. (2006). Maximum entropy modeling of species geographic distributions. *Ecological Modelling*, 190, 231–259.
- Qiao, H., Escobar, L. E., Saupe, E. E., Ji, L., & Soberón, J. (2017). A cautionary note on the use of hypervolume kernel density estimators in ecological niche modelling. *Global Ecology and Biogeography*, 26, 1066–1070.
- Qiao, H., Soberón, J., & Peterson, A. T. (2015). No silver bullets in correlative ecological niche modelling: Insights from testing among many potential algorithms for niche estimation. *Methods in Ecology and Evolution*, 6, 1126–1136.
- Schleuter, D., Daufresne, M., Massol, F., & Argillier, C. (2010). A user's guide to functional diversity indices. *Ecological Monographs*, 80, 469–484.
- Schölkopf, B., Platt, J. C., Shawe-Taylor, J., Smola, A. J., & Williamson, R. C. (2001). Estimating the support of a high-dimensional distribution. *Neural Computation*, 13, 1443–1471.
- Schölkopf, B., Williamson, R. C., Smola, A. J., Shawe-Taylor, J., & Platt, J. C. (1999). *Proceedings of the 12th International Conference on Neural Information Processing Systems*, pp. 582–588.
- Scott, D. W. (2015). *Multivariate density estimation: Theory, practice, and visualization*. New York, NY: John Wiley & Sons.
- Scott, D. W., & Wand, M. (1991). Feasibility of multivariate density estimates. *Biometrika*, 78, 197–205.
- Soberón, J., & Nakamura, M. (2009). Niches and distributional areas: Concepts, methods, and assumptions. *Proceedings of the National Academy of Sciences*, 106(Suppl 2), 19644–19650.
- Swanson, H. K., Lysy, M., Power, M., Stasko, A. D., Johnson, J. D., & Reist, J. D. (2015). A new probabilistic method for quantifying  $n$ -dimensional ecological niches and niche overlap. *Ecology*, 96, 318–324.
- Swenson, N. G., & Weiser, M. D. (2014). On the packing and filling of functional space in eastern North American tree assemblages. *Ecography*, 37, 1056–1062.
- Tervonen, T., van Valkenhoef, G., Baştürk, N., & Postmus, D. (2013). Hit-and-run enables efficient weight generation for simulation-based multiple criteria decision analysis. *European Journal of Operational Research*, 224, 552–559.
- Tingley, R., Vallinoto, M., Sequeira, F., & Kearney, M. R. (2014). Realized niche shift during a global biological invasion. *Proceedings of the National Academy of Sciences*, 111, 10233–10238.
- Villéger, S., Mason, N. W., & Mouillot, D. (2008). New multidimensional functional diversity indices for a multifaceted framework in functional ecology. *Ecology*, 89, 2290–2301.
- Violle, C., Thuiller, W., Mouquet, N., Munoz, F., Kraft, N. J. B., Cadotte, M. W., ... Mouillot, D. (2017). Functional rarity: The ecology of outliers. *Trends in Ecology & Evolution*, 32, 356–367.
- Wand, M., & Jones, M. (1994). Multivariate plug-in bandwidth selection. *Computational Statistics*, 9, 97–116.
- Westoby, M. (1998). A leaf-height-seed (LHS) plant ecology strategy scheme. *Plant and Soil*, 199, 213–227.

## SUPPORTING INFORMATION

Additional Supporting Information may be found online in the supporting information tab for this article.

**How to cite this article:** Blonder B, Morrow CB, Maitner B, et al. New approaches for delineating  $n$ -dimensional hypervolumes. *Methods Ecol Evol*. 2017;00:1–15. <https://doi.org/10.1111/2041-210X.12865>