BOUNDARY DETECTION FOR HIGH-DIMENSIONAL POINT CLOUDS

Anonymous authors
Paper under double-blind review

ABSTRACT

We introduce two new methods for boundary detection in high-dimensional point cloud data, intended to work in situations with non-uniform density, high ambient dimension, and substantial noise. Our first method fits a separating hyperplane on local neighborhoods after doing local dimension reduction—this method works best when one can reliably estimate the intrinsic dimension of the data. Our second method measures the directional uniformity of local neighborhoods, and has the best overall performance on synthetic data. In addition to pointwise numerical boundary scores, our algorithms yield an estimated normal vector at each boundary point. We validate our methodology on synthetic data as well as two real examples: we analyze spatial transcriptomics data, where we show that we can detect the boundary of a tumor, and we also study LLM activation geometry.

1 INTRODUCTION

The manifold hypothesis specifies that point cloud data is sampled from a distribution supported on a low-dimensional manifold. A natural variant of this setup is to assume that the support is instead a manifold with boundary. In this case, determining the location of the boundary and a normal vector to it become fundamental questions. Algorithms to determine the boundary have many applications. For example, in spatial transcriptomics data, each point on a grid has a gene expression vector that lies in $\mathbb{R}^n$ for n approximately 20000. Detecting boundary regions in genomic space allow us to determine qualitative changes in underlying tissue, for example transitions from cancerous tumor regions to healthy cells. Another application of interest is to study the geometry of LLM activations. Recent mechanistic interpretability work has discovered that residual-stream activations often lie on curved submanifolds rather than linear subspaces (Gurnee & Tegmark, 2024; Engels et al., 2025; Gurnee et al., 2026; Berdnikov & Liokumovich, 2026). Model behavior can be influenced by adding a "steering" vector at intermediate activations (Turner et al., 2023; Panickssery et al., 2023), but these interventions work best when one is able to stay on the underlying manifold (Wurgaft et al., 2026). Knowing the boundary points as well as having estimated boundary normals is useful for steering interventions and other interpretability problems. More broadly, boundary detection can be used for general-purpose methods such as clustering (Wang et al., 2018; Averbuch-Elor et al., 2020).

1.1 CONTRIBUTIONS

We introduce two general-purpose methods for identifying boundary points and calculating normal vectors at the boundary. Both methods are fast and practical for large data sets.

Our hyperplane method works the best among all tested methods when the local dimension can be reliably estimated. When the estimated dimension m is larger than the true dimension, it finds the correct boundary of the data as though it were m-dimensional. Our directional uniformity method is more robust when noise levels prevent accurate dimension estimates.

The hyperplane method is mathematically grounded by theorems proving that boundary points can be reliably distinguished from interior points in a way that scales well with intrinsic dimension. We empirically validate both methods on synthetic data.

Finally, we show that our methodology is effective on spatial transcriptomics data for identifying a tumor boundary, and we also apply it to find the boundary of LLM activations for the character-counting manifold of (Gurnee et al., 2026) for open-weight models.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1.2 Related Work

There are several boundary-detection methods that rely on the fact that boundary points have lower density than interior points. BORDER (Xia et al., 2006) counts the number of "reverse k nearest neighbors" (RkNN), i.e., the number of points for which the query point is a kNN. Boundary points tend to have fewer kNNs. Averbuch-Elor et al. (2020) and Du et al. (2021) also use reverse kNNs for boundary detection, as an intermediate step towards clustering. In more detail, Averbuch-Elor et al. (2020) and Du et al. (2021) calculate $\sum_{x_j \in \text{RkNN}(x_i)} K(x_i, x_j)$, where K is a Gaussian or Cauchy kernel, respectively. Reverse unreachability (RU) (Li et al., 2018) is different; it uses the representation matrix W from Roweis & Saul (2000) to approximate each point x_i as a linear combination $\sum_j W_{ij} x_j$ of its neighbors, and then reverse unreachability is defined as $\sum_j \mathbf{1}_{W_{ji} \leq 0}$. All of these methods perform poorly when the point cloud has non-uniform density.

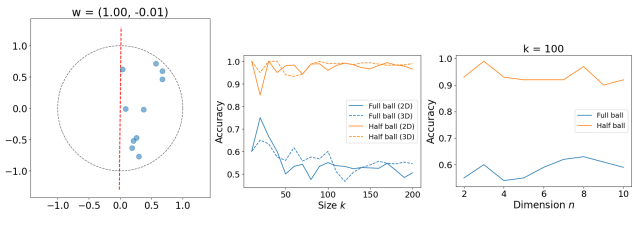
Some methods are only designed to be used for very low-dimensional data. The direction centrality measure (Peng et al., 2022) is defined for data in $\mathbb{R}^2$ and the level-set method (Ding et al., 2015) was only applied to 2D data. Both methods can be theoretically generalized to higher dimensions, but scale poorly with ambient dimension and are not practical for data that is more than two dimensional.

The last class of methods uses geometric structure to detection boundaries in data of any dimension. Border Edge Pattern Selection (BEPS) (Li & Maguire, 2011), CO (Wang et al., 2018), and LoDD (Peng et al., 2025) all start by projecting the nearest neighbors $\{x_{ij}\}$ to the unit sphere centered at the query point x_i. Let $\vec{y}_{ij} := (x_{ij} - x_i)/\left\|x_{ij} - x_i\right\|$ and define $\vec{v_i} := \frac{1}{k}\sum_{j=1}^{k} \vec{y}_{ij}$. BEPS is similar to our hyperplane method—the BEPS score $\frac{1}{k}\sum_j \mathbf{1}_{y_{ij} \cdot \vec{v_i} \geq 0}$ is the percentage of neighbors on the positive side of the hyperplane orthogonal to $\vec{v_i}$. It is far less robust to noise than either of our methods, though, because the direction of the vector $\vec{v_i}$ is sensitive to noisy perturbations off the manifold, especially on interior points where $\mathbb{E}[\vec{v_i}] = 0$.

The CO score (Wang et al., 2018) uses the same vector $\vec{v_i}$ but scales by density and measures the alignment between neighbors. At query point x_i, the score is $\sum_{j=1}^{k} m_i^{-1} m_j^{-1} \vec{v_i} \cdot \vec{v_j}$, where $m_i^{-1} = \sum_j \left\|x_{ij} - x_i\right\|$ is a "mass" term that is high in regions with low density. Like the other density-based methods, CO performs poorly on data with non-uniform density.

The LoDD score is $\omega(\sum_{i=1}^{d} \lambda_i)^2 + (1-\omega)\frac{2d}{d-1}\sum_i \sum_{j \neq i} \lambda_i \lambda_j$, where $\{\lambda_i\}$ is the set of eigenvalues of the covariance matrix for the centered, normalized neighbor vectors $\{\vec{y}_{ij}\}$ and $\omega \in [0, 1]$ is a hyperparameter that is set to .5 in Peng et al. (2025) and in our experiments in this paper. The LoDD score has a maximum of 1 that is achieved when the eigenvalues are equal and sum to 1. It is higher for interior points because $\{\vec{y}_{ij}\}$ is uniformly distributed on the d-dimensional sphere. LoDD has the strongest overall performance of all the existing methods we tested in the literature, but it is less robust than ours to noise, especially in high ambient dimensions. Unlike our methods (or BEPS or CO), LoDD does not produce normal vectors at each point—just pointwise boundary scores.

2 Hyperplane Method for Boundary Detection



(a) (b) (c)

Figure 1: Hyperplane optimization on points sampled from half and full balls. Accuracy is the percentage of points that lie on the positive side of the hyperplane orthogonal to $\vec{w}$.

The idea of our hyperplane (HP) boundary-detection algorithm, detailed in Algorithm 1, is that if x is a boundary point on an n-dimensional manifold, then it has a neighborhood that is diffeomorphic to an n-dimensional half ball. A half space is defined by a hyperplane in $\mathbb{R}^n$. If x is a boundary point, then it is possible to find a hyperplane such that most its neighbors lie on the same side of the hyperplane. On the other hand, if x is an interior point, then any hyperplane will split the neighborhood in half, with roughly half the points on one side and half on the other side.

We assign each point x a *boundary score* $s(x) \in [0, 1]$ that is the percentage of its neighborhood lying on the same side of an "optimal" hyperplane. Boundary points have scores near 1, and manifold points have scores near .5. Near a transverse intersection of manifolds, the score is usually higher than .5 but lower than 1—for example, near a transverse intersection of two manifolds, the score is near .75. (See Figure 6, for example.)[1] The algorithm steps are described below.

2.1 THE ALGORITHM

1. **Local PCA:** For each point x, we use local PCA to reduce the dimension of its local neighborhood $\{x_i\}_{i=1}^k$, where k is a parameter that controls the size of the neighborhood. If the dimension n of the manifold is known, then we reduce to n dimensions; otherwise, we use a spectral gap threshold condition. Let $\{\vec{z_i}\}_{i=1}^k$ be the neighborhood after dimension reduction and after centering so that x is mapped to $\vec{0}$.
2. **Hyperplane optimization:** We solve for an "optimal" hyperplane that best separates the data points onto a single side of the hyperplane. Every hyperplane is determined by an orthogonal vector $\vec{w}$ with norm $\|\vec{w}\| = 1$. A point $\vec{z_i}$ lies on the "positive" side of the hyperplane if $\vec{z_i} \cdot \vec{w} > 0$. A hyperplane is good if most of the points lie on the positive side of the hyperplane. To formulate this, we solve the optimization problem

$$\text{argmax}_{\vec{w}}\, f(\vec{w}) = \sum_{i=1}^{k} \vec{z_i} \cdot \vec{w}\,, \qquad \text{subject to } g(\vec{w}) = \|\vec{w}\|^2 = 1\,. \tag{1}$$

   The function $f(\vec{w})$ is proportional to the mean (signed) distance to the hyperplane, and the constraint normalizes the sum so that it cannot be arbitrarily large. By the method of Lagrange multipliers, the solution to our optimization problem is $\vec{w} = \sum_i \vec{z_i} / \left\|\sum_i \vec{z_i}\right\|$.
3. **Boundary score:** The boundary score $s(x) \in [0, 1]$ for point x is the percentage of neighboring points that lie on the positive side of the hyperplane. Boundary and interior points have scores close to 1 and .5, respectively.

As an alternative to our hyperplane optimization step, one could instead aim to maximize the number of points such that $\vec{z_i} \cdot \vec{w} \geq 0$. In contrast, maximizing $f(\vec{w})$ encourages but does not force points to lie on the positive hyperplane side. The advantage of our formulation is that it has a fast, closed-form solution. Algorithm 1 runs in about 1s for $N = 10,000$ points in $\mathbb{R}^{15}$.

Algorithm 1 Boundary Detection

Input: Point cloud $\mathcal{X}$, neighborhood size k, PCA spectral gap threshold τ (default $\tau = 7$)
Output: Boundary score $s(x) \in [0, 1]$ for each $x \in \mathcal{X}$

for all $x \in \mathcal{X}$ **do**
  $\mathcal{N} \leftarrow k$-nearest neighbors of x in $\mathcal{X}$
  $\{z_i\} \leftarrow \text{PCA}(\mathcal{N}, \tau)$ ▷ Use threshold τ to determine dimension
  $\{z_i\} \leftarrow \{z_i - z\}$ ▷ Center so projection of x is at origin
  $\vec{w} \leftarrow \dfrac{\sum_i \vec{z_i}}{\left\|\sum_i \vec{z_i}\right\|}$ ▷ $\vec{w}$ is orthogonal to an optimal hyperplane
  $m \leftarrow \left|\{z_i \mid \vec{z_i} \cdot \vec{w} \geq 0\}\right|$ ▷ Count neighbors on positive side of the hyperplane
  $s(x) \leftarrow m \,/\, k$ ▷ Divide by total size of the neighborhood
end for
return $\{s(x)\}_{x \in \mathcal{X}}$

To illustrate the hyperplane optimization step of the algorithm, we sample k points $\{\vec{z_i}\}$ from half or full n-dimensional balls and calculate $\vec{w}$. See Figure 1. For each experiment, we define the "accuracy" to be the percentage of points that lie on the positive side of the hyperplane determined by $\vec{w}$—this is analogous to the boundary score defined previously. When the points are sampled

[1] For example, suppose a point cloud is sampled from the intersection of the xy plane and the xz plane in $\mathbb{R}^3$. For a point on the xy plane with small but positive y coordinate, the optimal hyperplane P is one that is parallel to the xz plane with normal vector $\vec{w} = (0, -1, 0)$. The boundary score will be close to .75 because about half the points on the xy plane and *all* the points on the xz plane will be on the positive side of P.

from a half ball—which models the boundary-point case—the accuracy stays close to 1, even for the smallest neighborhood size that we tested ($k = 10$). By contrast, when the points are sampled from a full ball—which models the interior-point case—the accuracy is close to $1/2$. The gap between interior and boundary points persists as the dimension n increases to $n = 10$. This is preliminary evidence that the optimized hyperplane reliably distinguishes between boundary and interior points.

2.2 Concentration Inequalities for the Boundary Scores

The theorems below can be used to guide a choice of neighborhood size k given a manifold dimension n, or to pick a threshold on the boundary scores for identifying boundary points. Our goal is to quantify the distribution of boundary scores $s(x)$ when x is either an interior point or a boundary point. To do so, we consider the random variable

$$S_{nk} := \sum_{i=1}^{k} \mathbf{1}_{\vec{w}\cdot\vec{z}_i \geq 0}\,, \tag{2}$$

which counts the number of points that lie on the positive side of the hyperplane determined by $\vec{w}$. In our theoretical results below, we assume that the points $\{\vec{z}_i\}_{i=1}^k$ are drawn uniformly at random from either an n-dimensional ball (to model the situation where x is an interior point) or an n-dimensional half ball (to model the situation where x is a boundary point). The boundary score $s(x)$ behaves similarly to $\frac{1}{k}S_{nk}$, and we are interested in its probability distribution. The challenge is that the terms in the sum S_{nk} are generally not independent because $\vec{w}$ depends on all $\vec{z}_i$, so S_{nk} is not a binomial random variable. Below, we prove that boundary scores for boundary and interior points concentrate near 1 and $1/2$, respectively.

Theorem 2.1. *Let $\{\vec{z}_i\}_{i=1}^k$ be points sampled independently and uniformly at random from an n-dimensional ball. Then there are constants C_1, C_2 such that for any $\epsilon > 0$, we have*

$$\mathbb{P}\Big[|S_{nk}/k - 1/2| > \epsilon\Big] \leq \epsilon^{-2}\Big(C_1\sqrt{n/k} + C_2 n/k\Big)$$

for all dimensions n and all $k \geq 3$.

Theorem 2.2. *Let $\{\vec{z}_i\}_{i=1}^k$ be points sampled independently and uniformly at random from an n-dimensional half ball. For any $\epsilon > 0$, we have*

$$\mathbb{P}\Big[1 - S_{nk}/k \geq \epsilon\Big] \leq \frac{3}{\sqrt[3]{4}}\,\frac{(n+1)}{\epsilon\sqrt[3]{k}}\,.$$

3 Directional Uniformity Method for Boundary Detection

Our directional uniformity (DU) method is related to the Rayleigh uniformity test for deciding whether a spherical sample is uniform or not (Mardia & Jupp, 1999). Suppose that for each point x, we consider a local neighborhood $\{x_{ij}\}$ of size k. We project each neighbor onto a sphere centered at x and calculate the vector $\vec{v}_i := \frac{1}{k}\sum_j (x_j - x_i)/\big\|x_j - x_i\big\|$. If x is a manifold point, then the neighbors are uniformly distributed on the sphere, so $\mathbb{E}[\vec{v}_i] = 0$. If x is a boundary point, then $\mathbb{E}[\vec{v}_i]$ is nonzero and parallel to the boundary normal. We define the boundary score

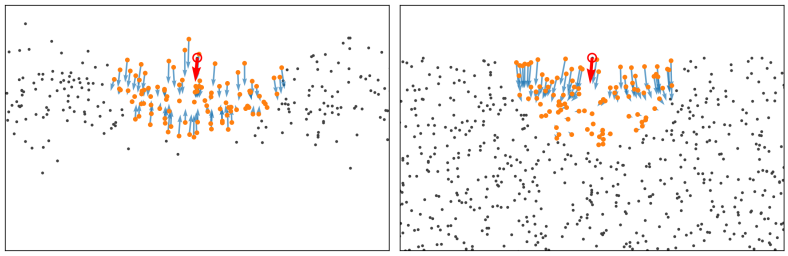


Figure 2: The vectors $\vec{v}_j$ for points $\{x_{ij}\}$ (orange) in a local neighborhood of a point x_i (red) within a larger point cloud that is sampled from (left) a thin strip in $\mathbb{R}^2$ with noise and (right) a half plane.

$$s(x) := \frac{1}{k^3}\sum_j \vec{v}_i \cdot \vec{v}_j\,. \tag{3}$$

Algorithm 2 Directional-Uniformity (DU) Boundary Detection

Input: Point cloud $\mathcal{X}$, neighborhood size k
Output: Boundary score $s(x) \in [0, 1]$ for each $x \in \mathcal{X}$

```
for all x ∈ X do
    N ← k-nearest neighbors of x in X
    v_i ← Σ_{j∈N} (x_j − x_i) / ||x_j − x_i||
    s(x) ← (1/k^3) Σ_j v_i · v_j
end for
return {s(x)}_{x∈X}
```

Note that expected boundary point scores are strictly less than 1 and depend on dimension n.

Robustness-to-noise is the main advantage of Equation (3) instead of simply $\|\vec{v_i}\|$. The former equation measures the alignment of $\vec{v_i}$ with its neighbors $\vec{v_j}$. Although there is always correlation between $\vec{v_i}$ and $\vec{v_j}$ due to the overlap in neighborhoods, there is stronger alignment when x is a boundary point because $\mathbb{E}[\vec{v_i}]$ is nonzero.

More importantly, ambient noise can thicken an n-dimensional manifold into a higher-dimensional object, and it is challenging for boundary detection methods to distinguish between the boundary of the ground-truth n-dimensional manifold and the boundary of the higher-dimensional object; Equation (3) helps to mitigate that. For example, in Figure 2, a 1D horizontal line segment in $\mathbb{R}^2$ has noise added so that it is a thin 2D subset. A point x_i at the top (or bottom) of the subset has a neighborhood that is genuinely a half ball, so $\|\vec{v_i}\|$ is as large as it is for a "true" boundary point. However, because it is a thin 2D strip, the points x_j at the bottom of the neighborhood are boundary points on the *opposite* side. For points x_j at the top and bottom, the vectors $\vec{v_j}$ point down and up, respectively. Therefore, there are cancellations in Equation (3) so that it is small, even though most of the vectors $\vec{v_j}$ are large. This correctly signifies that the point x_i is not on the true boundary (the left and right endpoints of the line segment). For comparison, the right panel of Figure 2 shows $\{\vec{v_j}\}$ for points x_j in a neighborhood of a point x_i on the boundary of a half plane. The norm $\vec{v_i}$ is close to that of the point on the line segment, but $\sum_j \vec{v_i} \cdot \vec{v_j}$ is larger. At a fixed neighborhood size and high noise level, Equation (3) is less likely than $\left\|v^i\right\|$ to flag interior points as boundary points.

4 Boundary Detection on Synthetic Data

Non-uniform density. Many methods in the literature erroneously detect low-density interior points as boundary points. To test this, we sample points non-uniformly from a 2D disk where there is a low-density region in the interior. For each method, we calculate AUC values (shown in Table 1) for boundary points vs. deep interior points (the low-density region).[2] See Figure 12 for the pointwise scores. RU (Li et al., 2018), BORDER (Xia et al., 2006), and BP (Averbuch-Elor et al., 2020; Du et al., 2021) explicitly rely on the idea that boundary points have lower density, so it is unsurprising that they have AUC values below .5. The CO method (Wang et al., 2018) performs similarly poorly (AUC .447) because of a "mass" term that increases the score at low-density points. The only methods that perform well on the non-uniform-density data set are LoDD, BEPS, our HP method, and our DU method, which all have AUC $= 1$.

DU (us)	Hyperplane (us)	LoDD	BEPS	RU	CO	BORDER	BP (Gauss.)	BP (Cauchy)
1.000	1.000	1.000	1.000	0.474	0.447	0.474	0.425	0.436

Table 1: Boundary ($\|x\| > .98$) vs deep-interior ($\|x\| < .23$) AUC for $N = 10,000$ points sampled non-uniformly from a 2D disk. In the interior, there is a "well" ($r < .23$) where the density is $1/9$ that of the density in the outer region ($r > .67$). The density increases linearly in between.

[2]The ground-truth disk boundary is the set of points x where $\|x\| = 1$. This boundary has measure zero, so zero of the sampled points are on the boundary. Instead, we conservatively define the empirical boundary to be the set of points with $\|x\| > .98$. In later sections, we lower this threshold to .95 with justification given in Section C.2, where we also show the results for other threshold choices.

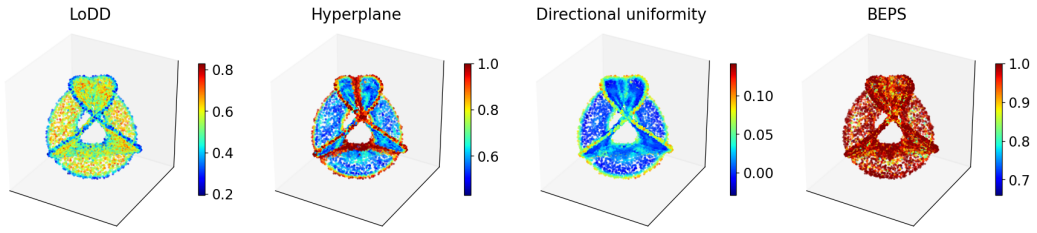


Figure 4: Boundary scores ($N = 10,000$, $k = 100$) on the Henneberg minimal surface embedded in $\mathbb{R}^{15}$ with isotropic Gaussian noise ($\sigma = .02$), plotted in $\mathbb{R}^3$.

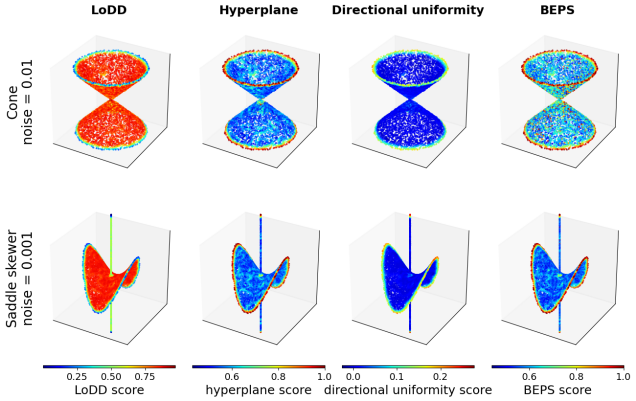


Figure 3: Boundary scores ($k = 100$) for two synthetic data sets ($N = 10^4$) with low noise levels.

Surfaces with non-trivial geometry. For the remainder of Section 4, our primary focus is comparing LoDD (Peng et al., 2025) and BEPS (Li & Maguire, 2011) with our HP and DU methods. We test the four methods on six synthetic data sets with low levels of Gaussian noise; see Figure 3 for the cone and saddle-skewer, and Figure 13 for four more. BEPS performs reasonably well, but it has a high number of false positives (interior points that score as high as boundary points) on three of the surfaces (the cone, Saturn, and the three disks). LoDD struggles with data points that have intrinsic dimension one, such as the skewer in the saddle-skewer. The interior 1D points have LoDD scores that are as low as points near the surface boundaries. (LoDD scores are close to 0 on boundary points and higher on the interior, unlike the other three methods.) HP scores boundary points close to 1, while manifold points have low boundary scores close to .5. Points near (but not on) a transverse intersection of two surfaces have moderate hyperplane boundary scores near .75. Similarly, DU scores are higher at boundary points and lower (close to 0) at interior points. This data set shows that all four methods are reasonably accurate on surfaces with complex topologies and curvature, except that LoDD confuses interior 1D points with boundary points and BEPS is prone to false positives, sometimes confusing isolated interior points for boundary points.

High ambient dimension. In high ambient dimension with low noise, BEPS cannot distinguish between boundary points and interior points, and LoDD scores have high variance on interior points. For example, see Figure 4. BEPS scores are roughly constant across all points, and many interior points have LoDD scores that are as low as boundary-point LoDD scores. In contrast, our HP and DU methods both highlight a clear boundary, scoring interior points much lower, although HP has a small amount of interior noise. A limitation of HP in high ambient dimensions is that it scores self-intersections as high as boundary points. (In lower ambient dimensions, HP assigns score $\approx .75 < 1$ to points near intersections.) For the remainder of this subsection, we focus on HP, DU, and LoDD.

Next, we study 2D disks embedded in $\mathbb{R}^D$ ($3 \leq D \leq 15$) with isotropic Gaussian noise ($\sigma = .01$). In Figure 5(a), we calculate AUC values for interior vs. boundary points[3]. Both the hyperplane and DU methods have AUC values near 1 at all D, while LoDD's performance degrades as D increases. That is, HP and DU perform better than LoDD on point clouds with high ambient dimension.

[3]We define the empirical boundary to be the set of points with $\|x\| > .95$, and the empirical interior to be the set of points with $\|x\| < .8$. We ignore the points with norms in between because their classification is ambiguous. See Section C.2 for more details on how we derived our definitions of interior and boundary points and Figure 20 for AUC values for alternative choices.

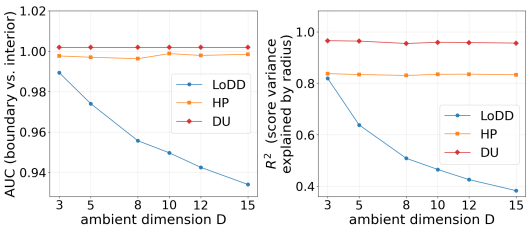


Figure 5: Analysis of boundary scores on disks embedded in $\mathbb{R}^D$ with Gaussian noise $\sigma = .01$ (size $N = 10,000$ points, with $k = 100$).

For each method (excluding BEPS), we measure the strength of the relationship between a point's norm and its boundary score. A good method should score all interior points at an approximately constant value and then monotonically change as one approaches the boundary. For each method we fit a monotonic function $f(r)$ to the scores as a function of radius r and calculate the coefficient of determination R^2. Figure 5(b) shows that the DU method has the highest R^2 value (near 1) at each D, while LoDD has the lowest values, reflecting its high variance within the interior of the disk.

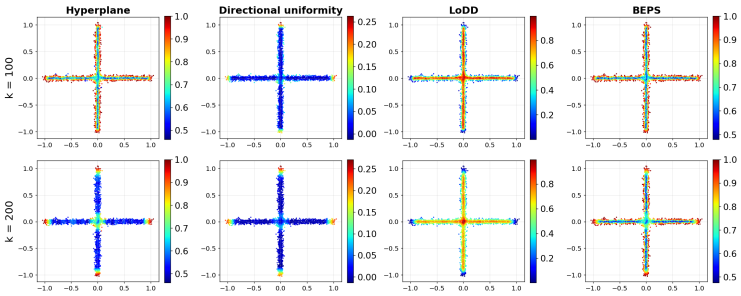


Figure 6: Boundary scores on noisy intersecting line segments.

High noise levels. At higher levels of noise, local PCA may overestimate the local dimension. This affects HP. If the estimated dimension m is larger than the true manifold dimension n, then HP will find the boundary of it *as an m-dimensional object* rather than an n-dimensional object. In Figure 6, we show an example where a point cloud ($N = 2,500$) is sampled from two intersecting orthogonal line segments with Gaussian noise $\sigma = .025$. At this noise level, with default neighborhood size $k = 100$, local PCA over-estimates the intrinsic dimension—99.1% of points are assigned local dimension 2. HP thus detects the edges of the thickened arms as boundary points, because this is the boundary when the object is treated as 2-dimensional. Increasing the neighborhood size to $k = 200$ resolves the issue because at the larger neighborhood size, local PCA correctly estimates dimension 1 at most points except near the intersection. Interestingly, LoDD and BEPS suffer from the same limitation, but increasing the neighborhood size does not help them. DU is able to correctly identify true boundary points at both neighborhood sizes.

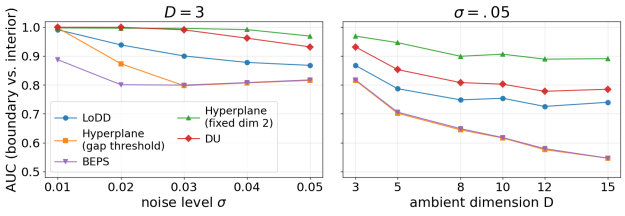


Figure 7: Boundary scores at high noise, for 2D disks ($N = 10,000$) embedded in $\mathbb{R}^D$ with Gaussian noise σ. We use neighborhood size $k = 100$. Boundary and interior points are defined as points with norm $> .95$ and $< .8$, respectively, with norm computed in the xy plane.

In Figure 7, we study each method's boundary-detection performance as we increase the noise level σ. The point clouds are sampled from 2D disks embedded in $\mathbb{R}^3$ with isotropic Gaussian noise. Again, we see that at higher noise levels, HP often misclassifies interior points as boundary points because local PCA overestimates the dimension as 3 instead of 2. An interior point with high $|z|$ coordinate is seen as the "boundary" of the thickened disk. However, if we fix dimension 2 for the local PCA dimension, then HP is able to correctly recover the disk boundary. We plot the AUC values as a function of σ in Figure 7(a). At the highest noise level ($\sigma = .05$), HP with fixed dimension (AUC .969) performs best, followed by DU (AUC .932), then LoDD (AUC .868), and then finally BEPS (AUC .818) and HP with the spectral-gap threshold condition (AUC .816). In the high-noise experiments, BEPS and HP (spectral gap threshold) are almost identical because both methods optimize for a hyperplane (in different ways) that maximizes the number of points on a single side of the hyperplane. When the local dimension is

overestimated, HP suffers from the same limitations as BEPS. We see the same behavior when the 2D disk is embedded in higher-dimensional $\mathbb{R}^D$ with noise level $\sigma = .05$, in Figure 7(b). See Figure 16 for scatter plots of the boundary scores at $D = 3$ and $\sigma = .05$. This suggests that HP should be used if the intrinsic dimension can be reliably estimated, but otherwise, it is best to use DU.

High intrinsic dimension. All methods (hyperplane, DU, LoDD, and BEPS) are able to detect the boundary of an n-dimensional object. We sample $N = 10,000$ points from n-balls embedded in $\mathbb{R}^n$ for $n = 3, 5, 7, 10$. For each method, the AUC value is 1 for each n, where boundary and interior points are points with norm $> .95$ and $< .8$, respectively. See Figure 17 for boundary scores.

Boundary normals. Both of our boundary detection methods yield normal directions at each point, in addition to numerical scores. For HP, we take the vector $\vec{w}$ (line 5 of Algorithm 1) and lift it from its PCA coordinates back into the ambient space to get a normal $\vec{w}_{\text{amb}}$. For DU, the vector $\vec{v}_i := \frac{1}{k}\sum_j (x_{ij} - x_i)/\|x_{ij} - x_i\|$ is the same as the vector used by BEPS (Li & Maguire, 2011) and CO (Wang et al., 2018). In Figure 8, we test the accuracy of these vectors on 2D disks and n-balls. For 2D disks in $\mathbb{R}^3$, HP generally yields a more accurate normal direction estimate than DU, with the performance gap narrowing as the noise σ increases. For HP, fixing the dimension 2 works better than using a spectral gap threshold condition at lower noise levels ($\sigma \leq .05$), but at the highest noise level ($\sigma = .1$), the top two local principal components do not align as well with the tangent space. For 2D disks in $\mathbb{R}^D$, the hyperplane normal $\vec{w}_{\text{amb}}$ is significantly more accurate than $\vec{v}_i$ as D increases. For n-balls, both methods (and variants) are equally accurate; we suspect that this is usually true for manifolds with codimension zero. Overall, HP generally yields a much more accurate boundary normal direction compared to DU.

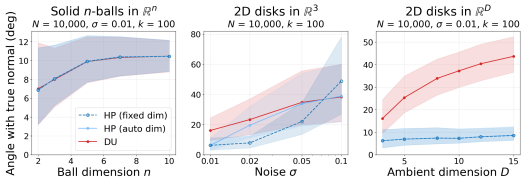


Figure 8: For points x near the boundary of a 2D disk or n-ball ($\|x\| > .95$), we compare the estimated boundary normal with the ground-truth normal $-x/\|x\|$. We plot median error and IQR.

5 Boundary Detection in Spatial Transcriptomics Data

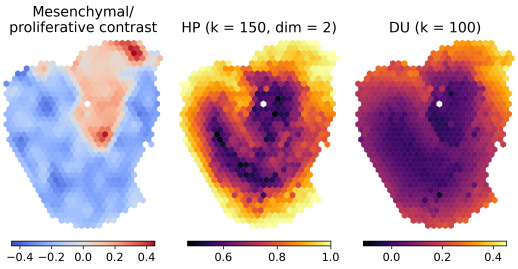
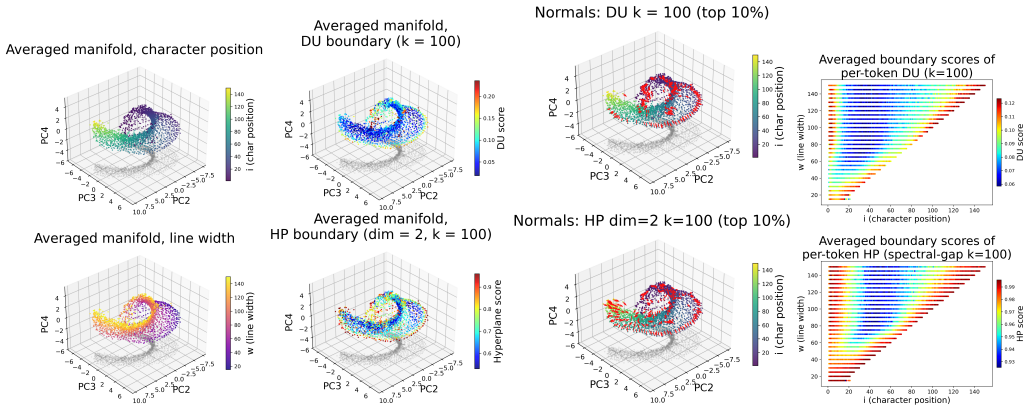


Figure 9: Glioblastoma slice ST1A1.

As a first application, we apply our methods to spatial transcriptomics data, which measures gene expression across a tissue section, to recover cell-state boundaries in glioblastoma. Glioblastoma is an aggressive brain tumor that diffusely infiltrates the brain, making it difficult to distinguish regions where tumor cells are abundant or sparse. Malignant cells also occupy distinct transcriptional states (Neftel et al., 2019) that may be amenable to different therapies, further motivating the identification of boundaries within the tumor. We analyze tissue slice ST1A1 from Al-Dalahmah et al. (2023).

Each capture area measures the combined expression of multiple cells. To interpret our boundary scores, we use published cell-state abundance estimates obtained by Robust Cell Type Decomposition (RCTD) (Cable et al., 2022), which deconvolves this expression into contributions from different cell states. Following Al-Dalahmah et al. (2023), we combine the malignant-cell subpopulations gl_Mes1 and gl_Mes2 into an astrocyte-like/mesenchymal group (Mes), and gl_Pro1 and gl_Pro2 into a proliferative group (Pro). These represent distinct malignant states that have prognostic and therapeutic implications. Figure 9 shows their spatial distributions and the mesenchymal/proliferative contrast, calculated as Mes minus Pro.

In Figure 9, both HP and DU assign higher boundary scores along the tissue edge and near parts of the transition between Mes- and Pro-enriched regions. After pre-processing (see Section C.3), we reduce gene expression to 30 principal components and concatenate these with the two spatial



(a) (b) (c) (d)

Figure 10: Boundary detection on the character-counting manifold for Pythia-2.8B.

coordinates, yielding a point cloud of size $N = 741$ (the number of spatial spots) in $\mathbb{R}^{32}$. We view this as the graph of an expression function over spatial coordinates, motivating fixed dimension two for HP. For both methods, we construct each spot's neighborhood by taking the intersection of nearest neighbors with respect to both spatial coordinates and gene expression coordinates. Our results suggest that our methods can detect internal transitions between tumor states without using cell-state labels or predefined tumor-marker genes.

6 LLM Activation Geometry

As as second application, we reproduce the character-counting manifold of Gurnee et al. (2026) on two open-weight models (Pythia-2.8B and Llama-3.1-8B) and apply our boundary-detection methods. Using text from the C4 (English) data set (Raffel et al., 2020), we strip newlines and then re-insert them every k characters, rounded to the nearest word boundary, for $k \in \{15, 20, \ldots, 150\}$. This produces 28 versions of each document, each with fixed line width w. Every non-newline token has a character position i within its line of width w. We extract the layer-4 residual-stream activations and average the activations for each (i, w), yielding point clouds of size $N \approx 2,300$ for each model. Figure 10 shows the Pythia-2.8B point cloud; see Figure 19 for Llama-3.1-8B. See Section C.4 for details of our procedure.

In Figure 10(b), both of our boundary detectors correctly assign higher scores to the ground-truth boundary of the averaged manifold, which is $\{(i, w) \mid i = 0 \text{ or } i = w \text{ or } w = 15 \text{ or } w = 150\}$. For HP, we use dimension $n = 2$ because the surface is intrinsically parameterized by character position i and line width w. We also show the vector fields of estimated boundary normals in Figure 10(c) for the top 10% scoring points; the vectors tend to point inwards on the surface, as they should.

Lastly, we test HP and DU on the un-averaged, per-token point cloud (size $N \approx 250$k per model). See Figure 10(d). This is much more challenging because the intrinsic dimension of the space is much higher (and unknown), and there are other, unknown boundaries in the token point cloud. After calculating boundary scores at each point, we average the boundary scores for the tokens at each (i, w). The averaged boundary scores are consistently higher at the ground-truth boundary.

7 Conclusions

We presented two new methods for identifying boundary points and calculating boundary normal vectors in high-dimensional, noisy data. We justified the hyperplane method mathematically, and we rigorously tested both methods empirically against others in the literature on synthetic data. Finally, we showed that our boundary detection methods are effective on real-world applications by using them to identify a tumor boundary in a spatial transcriptomics data set and to characterize the boundary of a manifold in LLM activations.

AI USE STATEMENT

In this work, we used generative AI tools to

- write almost all of the code, including the code to implement our methods, generate (some) of the synthetic data sets, compare our methods against others in the literature, generate the figures, and run the spatial transcriptomics and LLM experiments,
- assist with isolated steps within the proofs of lemmas, particularly for some of the proofs in Section B.1 for Theorem 2.1,
- help design the methodology for reproducing the character-counting manifold, although the idea for using it as an example was our own,
- assist with very minor research methodology choices, such as the decision in Figure 5 to fit a monotonic function to the boundary scores for evaluating performance,
- assist with finding related papers for us to read,
- format bibtex references.

We have not used generative AI tools for

- developing theoretical models or conceptual frameworks,
- writing the paper, including writing up the mathematical proofs,
- interpreting the results,
- qualitative and thematic data analysis
- formulating the statements of our theorems or lemmas, or the high-level structure of the proofs
- cleaning and reformatting datasets

Assisting with translation is not applicable to this work.

We have reviewed all AI-assisted work. In particular, we have reviewed the code and checked all mathematical arguments for correctness. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We have released the source code and scripts for reproducing the figures and experiments[4]. Methodological details for the experiments are described in Section C.

REFERENCES

Osama Al-Dalahmah, Michael G. Argenziano, Adithya Kannan, Aayushi Mahajan, Julia Furnari, Fahad Paryani, Deborah Boyett, Akshay Save, Nelson Humala, Fatima Khan, Juncheng Li, Hong Lu, Yu Sun, John F. Tuddenham, Alexander R. Goldberg, Athanassios Dovas, Matei A. Banu, Tejaswi Sudhakar, Erin Bush, Andrew B. Lassman, Guy M. McKhann, Brian J. A. Gill, Brett Youngerman, Michael B. Sisti, Jeffrey N. Bruce, Peter A. Sims, Vilas Menon, and Peter Canoll. Re-convolving the compositional landscape of primary and recurrent glioblastoma reveals prognostic and targetable tissue states. *Nature Communications*, 14(1):2586, 2023. doi: 10.1038/s41467-023-38186-1.

Hadar Averbuch-Elor, Nadav Bar, and Daniel Cohen-Or. Border-peeling clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(7):1791–1797, 2020. ISSN 1939-3539.

Aleksandr Berdnikov and Yevgeny Liokumovich. Sky sphere representation in language models. *arXiv:2607.27092*, 2026.

[4] https://anonymous.4open.science/r/boundary_detection-B58D/

Dylan M. Cable, Evan Murray, Luli S. Zou, Aleksandrina Goeva, Evan Z. Macosko, Fei Chen, and Rafael A. Irizarry. Robust decomposition of cell type mixtures in spatial transcriptomics. *Nature Biotechnology*, 40(4):517–526, 2022. doi: 10.1038/s41587-021-00830-w.

Xuemei Ding, Yuhua Li, Ammar Belatreche, and Liam P. Maguire. Novelty detection using level set methods. *IEEE Transactions on Neural Networks and Learning Systems*, 26(3):576–588, 2015. ISSN 2162-2388.

Mingjing Du, Ru Wang, Ru Ji, Xia Wang, and Yongquan Dong. ROBP a robust border-peeling clustering using Cauchy kernel. *Information Sciences*, 571:375–400, 2021. doi: 10.1016/j.ins.2021.04.089.

Joshua Engels, Isaac Liao, Eric J. Michaud, Wes Gurnee, and Max Tegmark. Not all language model features are linear. In *International Conference on Learning Representations (ICLR)*, 2025.

Walter Gautschi. Some Elementary Inequalities Relating to the Gamma and Incomplete Gamma Function. *Journal of Mathematics and Physics*, 38(1-4):77–81, 1959. ISSN 0097-1421.

Wes Gurnee and Max Tegmark. Language models represent space and time. In *International Conference on Learning Representations (ICLR)*, 2024.

Wes Gurnee, Emmanuel Ameisen, Isaac Kauvar, Julius Tarng, Adam Pearce, Chris Olah, and Joshua Batson. When models manipulate manifolds: The geometry of a counting task. *arXiv:2601.04480*, 2026.

Xiaojie Li, Jiancheng Lv, and Zhang Yi. An Efficient Representation-Based Method for Boundary Point and Outlier Detection. *IEEE Transactions on Neural Networks and Learning Systems*, 29 (1):51–62, 2018. ISSN 2162-2388.

Yuhua Li and Liam Maguire. Selecting Critical Patterns Based on Local Geometrical and Statistical Information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(6):1189–1201, 2011. ISSN 1939-3539.

Kanti V. Mardia and Peter E. Jupp. *Directional Statistics*. Wiley Series in Probability and Statistics. Wiley, 1 edition, 1999. ISBN 9780471953333 9780470316979.

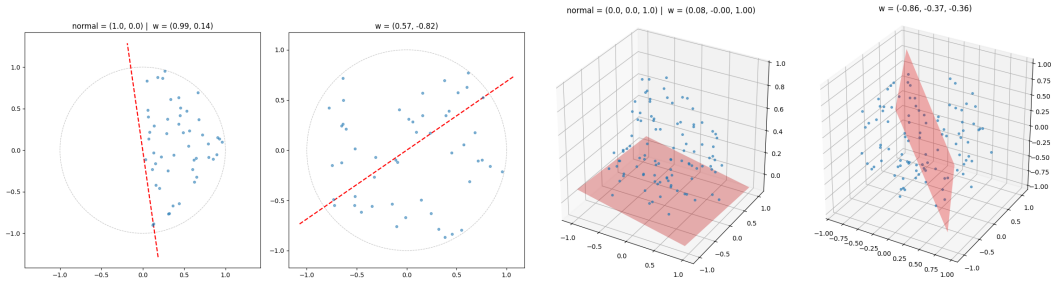
Cyril Neftel, Julie Laffy, Mariella G. Filbin, Toshiro Hara, Marni E. Shore, Gilbert J. Rahme, Alyssa R. Richman, Dana Silverbush, McKenzie L. Shaw, Christine M. Hebert, John Dewitt, Simon Gritsch, Elizabeth M. Perez, L. Nicolas Gonzalez Castro, Xiaoyang Lan, Nicholas Druck, Christopher Rodman, Danielle Dionne, Alexander Kaplan, Mia S. Bertalan, Julia Small, Kristine Pelton, Sarah Becker, Dennis Bonal, Quang-De Nguyen, Rachel L. Servis, Jeremy M. Fung, Ravindra Mylvaganam, Lisa Mayr, Johannes Gojo, Christine Haberler, Rene Geyeregger, Thomas Czech, Irene Slavc, Brian V. Nahed, William T. Curry, Bob S. Carter, Hiroaki Wakimoto, Priscilla K. Brastianos, Tracy T. Batchelor, Anat Stemmer-Rachamimov, Maria Martinez-Lage, Matthew P. Frosch, Ivan Stamenkovic, Nicolo Riggi, Esther Rheinbay, Michelle Monje, Orit Rozenblatt-Rosen, Daniel P. Cahill, Anoop P. Patel, Tony Hunter, Inder M. Verma, Keith L. Ligon, David N. Louis, Aviv Regev, Bradley E. Bernstein, Itay Tirosh, and Mario L. Suvà. An Integrative Model of Cellular States, Plasticity, and Genetics for Glioblastoma. *Cell*, 178(4): 835–849.e21, 2019. doi: 10.1016/j.cell.2019.06.024.

Nina Panickssery, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Matt Turner. Steering llama 2 via contrastive activation addition. *arXiv:2312.06681*, 2023.

Dehua Peng, Zhipeng Gui, Dehe Wang, Yuncheng Ma, Zichen Huang, Yu Zhou, and Huayi Wu. Clustering by measuring local direction centrality for data with heterogeneous density and weak connectivity. *Nature Communications*, 13(1):5455, 2022. ISSN 2041-1723.

Dehua Peng, Zhipeng Gui, Jie Gui, and Huayi Wu. A Robust and Efficient Boundary Point Detection Method by Measuring Local Direction Dispersion. *IEEE Transactions on Circuits and Systems for Video Technology*, 35(8):7409–7422, August 2025. ISSN 1558-2205.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.



(a) (b) (c) (d)

Figure 11: We calculate the vector $\vec{w}$ that is the solution to Equation (1) in our hyperplane method, where the accuracy of $\vec{w}$ is the percentage of points $\{z_i\}$ such that $\vec{w} \cdot \vec{z}_i \geq 0$. (A) 2D half ball with normal $(1, 0)$ and $k = 50$ points, with accuracy .98, (B) 2D full ball with $k = 50$ and accuracy .5, (C) 3D half ball with normal $(0, 0, 1)$ and $k = 100$, with accuracy .99, and (D) 3D full ball with $k = 100$ and accuracy .6.

Sam T. Roweis and Lawrence K. Saul. Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science*, 290(5500):2323–2326, 2000.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering language models with activation engineering. *arXiv:2308.10248*, 2023.

Zhiqiang Wang, Zhiwen Yu, C. L. Philip Chen, Jane You, Tianlong Gu, Hau-San Wong, and Jun Zhang. Clustering by local gravitation. *IEEE Transactions on Cybernetics*, 48(5):1383–1396, 2018. ISSN 2168-2275.

Daniel Wurgaft, Can Rager, Matthew Kowal, Vasudev Shyam, Sheridan Feucht, Usha Bhalla, Tal Haklay, Eric Bigelow, Raphael Sarfati, Thomas McGrath, Owen Lewis, Jack Merullo, Noah Goodman, Thomas Fel, Atticus Geiger, and Ekdeep Singh Lubana. Manifold steering reveals the shared geometry of neural network representation and behavior. *arXiv:2605.05115*, 2026.

Chenyi Xia, W. Hsu, M.L. Lee, and B.C. Ooi. BORDER: Efficient computation of boundary points. *IEEE Transactions on Knowledge and Data Engineering*, 18(3):289–303, 2006. ISSN 1558-2191.

A Additional Experimental Figures

A.1 Hyperplane optimization in half and full balls

To illustrate the intuition for HP, we calculate the vector $\vec{w}$, the solution to Equation (1), for points sampled from 2D and 3D balls and half balls. In Figure 1(a), we showed $\vec{w}$ and its orthogonal hyperplane (line) for $k = 10$ points sampled from a 2D half ball. In Figure 11, we show $\vec{w}$ and its orthogonal hyperplane for $k = 50$ points in 2D half and full balls and for $k = 100$ points in 3D half and full balls. In all cases, the "accuracy" (the percentage of points $\vec{z}$ such that $\vec{w} \cdot \vec{z} \geq 0$) is close to 1 for the half balls and close to .5 for the full balls.

A.2 Non-uniform density

In Section 4, we tested boundary detection methods on a 2D disk with non-uniform density and reported AUC values in Table 1. In Figure 12, we show scatter plots of the pointwise boundary scores for each method.

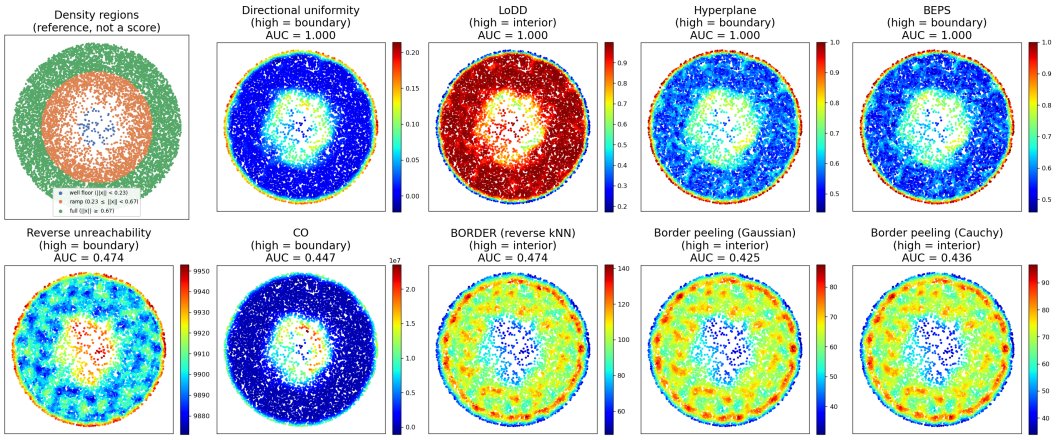


Figure 12: We sample $N = 10,000$ points non-uniformly from a 2D disk. In the interior, there is a "well" ($r < .23$) where the density is $1/9$ that of the density in the outer region ($r > .67$). The density increases linearly in between. For each method, we calculate the AUC value for boundary points ($\|x\| > .98$) vs. interior well points ($\|x\| < .23$), using neighborhood size $k = 100$. In the top row, we show the methods that are able to handle non-uniform density. In the bottom row, we show the methods that confuse boundary points and low-density interior points.

A.3 Surfaces with non-trivial geometry

In Section 4, we calculated boundary scores for the hyperplane method, DU, LoDD, and BEPS on a cone and a saddle-skewer data set with low noise levels. In Figure 13 below, we show the boundary scores for four more synthetic data sets.

A.4 High ambient dimension

In Section 4, we calculated boundary scores for the hyperplane method, DU, LoDD, and BEPS for surfaces in high ambient dimension with Gaussian noise. Figure 4 showed the Henneberg minimal surface in $\mathbb{R}^{15}$ and Figure 5 analyzed the boundary scores for disks in $\mathbb{R}^D$. In Figure 14, we show the boundary scores for three more surfaces in $\mathbb{R}^{15}$, and in Figure 15, we show the boundary scores for 2D disks in $\mathbb{R}^{15}$ with noise.

A.5 High noise levels

In Section 4, we tested our hyperplane and DU methods against LoDD and BEPS on 2D disks embedded in $\mathbb{R}^D$ with increasing noise level σ. In Figure 16, we show the boundary scores for disks in R^3 at the highest noise level $\sigma = .05$.

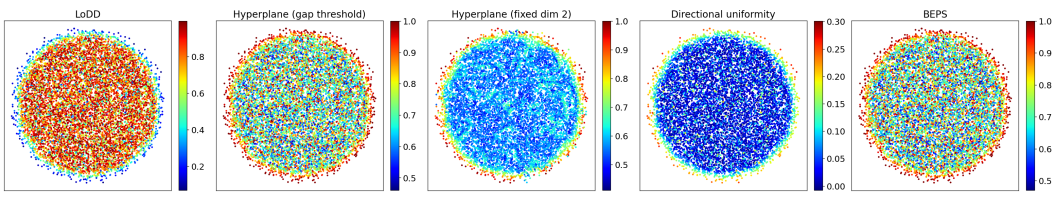


Figure 16: Boundary scores for 2D disks embedded in $\mathbb{R}^3$ with isotropic Gaussian noise $\sigma = .05$, projected here to $\mathbb{R}^2$.

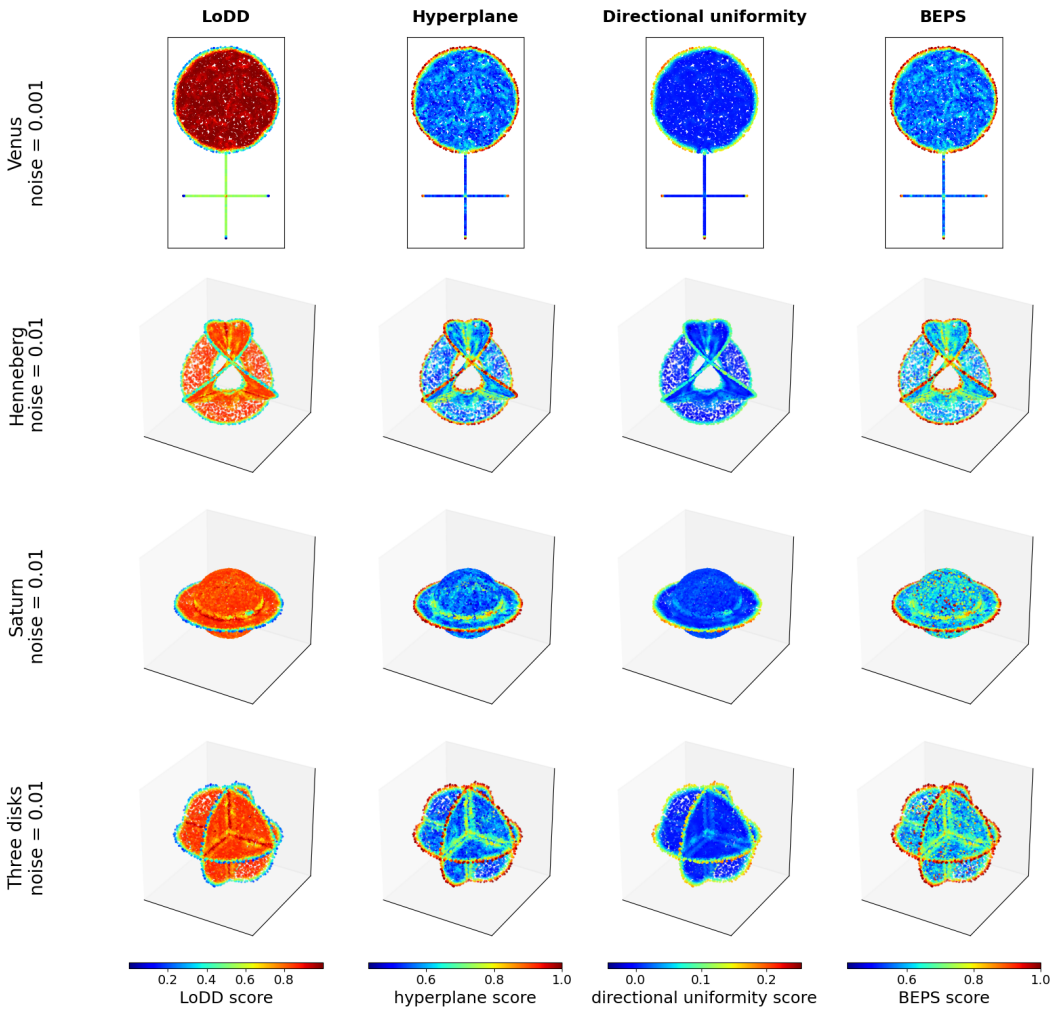


Figure 13: Boundary scores for four synthetic data sets with low noise levels. Each point cloud has $N = 10,000$ points, and we use neighborhood size $k = 100$.

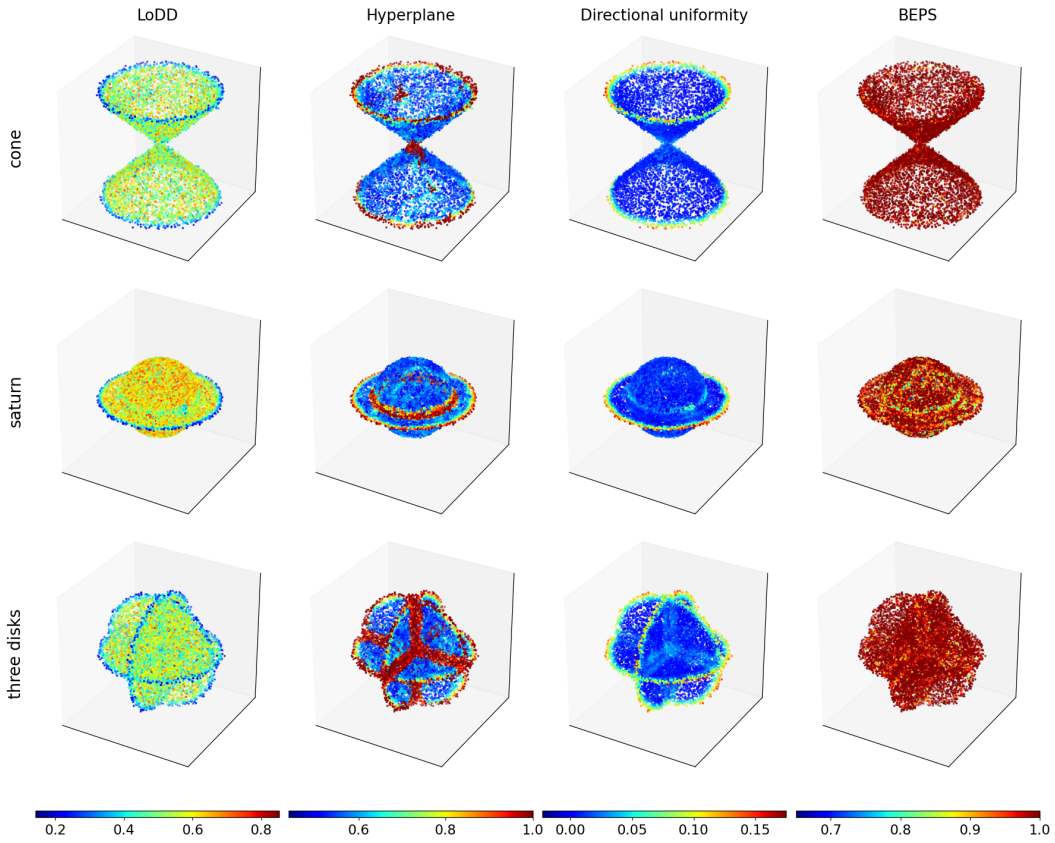


Figure 14: Boundary scores for three synthetic data sets embedded in $\mathbb{R}^{15}$ with noise. Scores here are plotted in $\mathbb{R}^3$.

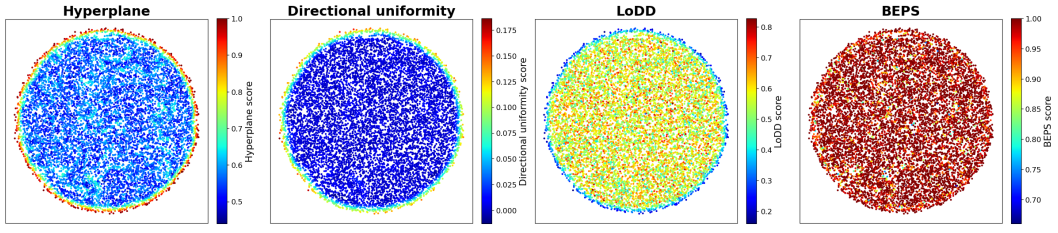


Figure 15: Boundary scores on disks embedded in $\mathbb{R}^{15}$ with Gaussian noise $\sigma = .01$, with $N = 10,000$ points. We use neighborhood size $k = 100$. We plot only the first two coordinates.

A.6 HIGH INTRINSIC DIMENSION

In Figure 17, we show the boundary scores for points sampled from a 10-dimensional ball in $\mathbb{R}^{10}$. All methods achieve AUC 1 except for reverse unreachability (AUC .01). For the AUC calculation, we define boundary and interior points as points with norm $> .95$ and $.8$, respectively.

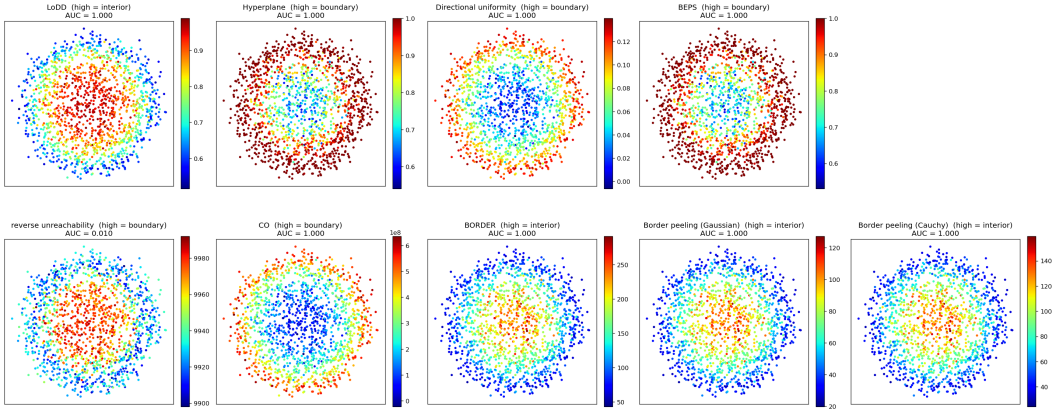


Figure 17: Boundary scores on a point cloud with $N = 10,000$ points sampled from a 10-dimensional ball in $\mathbb{R}^{10}$, with isotropic Gaussian noise ($\sigma = .01$). The ground-truth boundary is difficult to display in 2D because it is 9-dimensional, so instead we take a thin 2D slice (a slightly thickened disk) the ball, plotting the 1500 points closest to the xy plane. Because of the curvature of the boundary (a sphere), the boundary appears thicker than the 2D balls in earlier figures, but this is an artifact of the way we are taking a slice and projecting to $\mathbb{R}^2$.

A.7 SPATIAL TRANSCRIPTOMICS

In Section 5, we analyzed glioblastoma slice ST1A1 from Al-Dalahmah et al. (2023). In Figure 9, we showed HP and DU scores at neighborhood sizes $k = 150$ and $k = 100$, respectively. In Figure 18 below, we show the boundary scores at neighborhood sizes $k = 75, 100, 150$ for each method.

A.8 LLAMA-3.1-8B ACTIVATION GEOMETRY

We reproduce the character-counting manifold of (Gurnee et al., 2026) for Llama-3.1-8B using the methodology in Section C.4. In Figure 19, we show the averaged manifold and our detected boundaries. We also show the average boundary scores at each character position i and line width w when we compute boundary scores for the un-averaged, per-token point cloud. See Figure 10 for Pythia-2.8B.

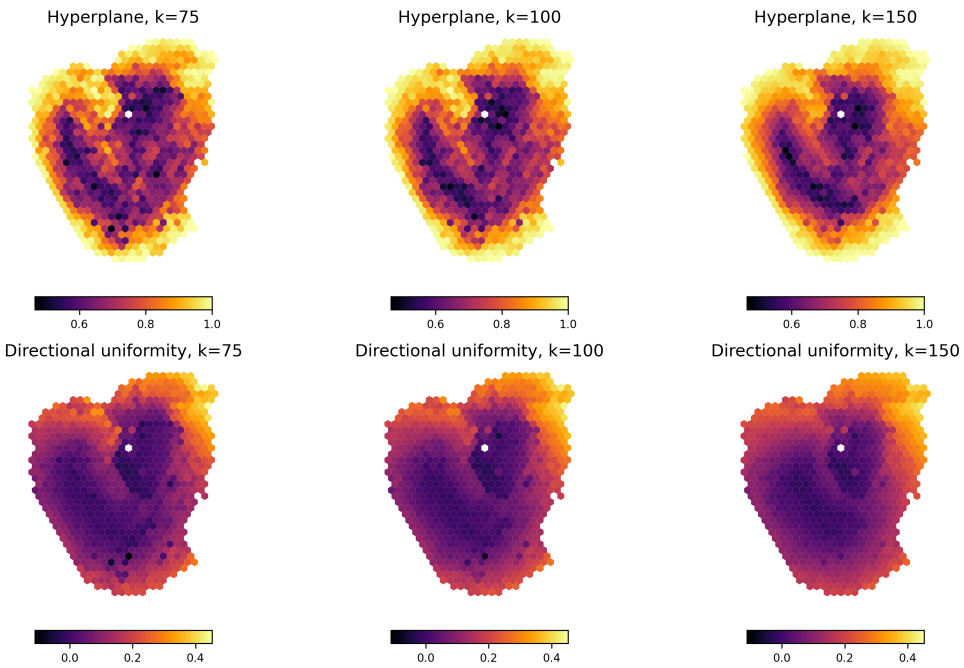
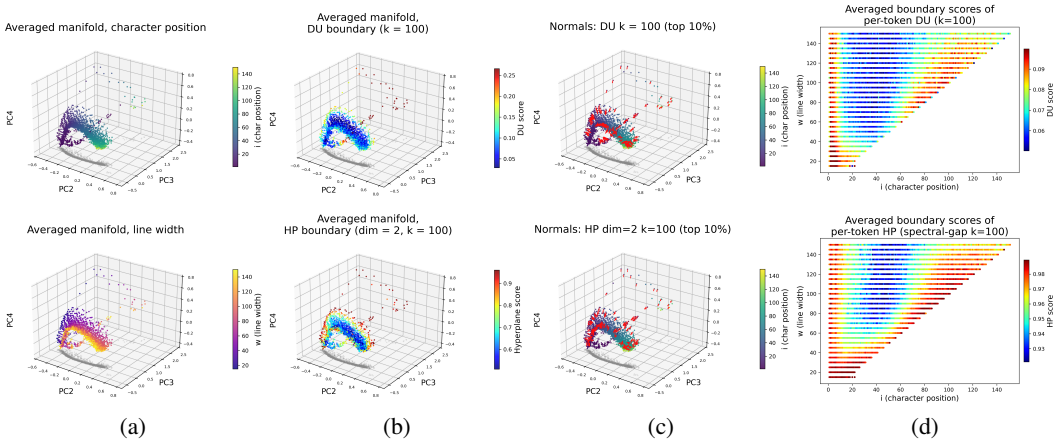


Figure 18: HP and DU boundary scores on the ST1A1 glioblastoma slice at neighborhood sizes $k = 75, 100, 150$.



(a) (b) (c) (d)

Figure 19: Boundary detection on the character-counting manifold for Llama-3.1-8B. In (c), we calculate normal vectors only for the top 10% scoring boundary points.

B MATHEMATICAL PROOFS

B.1 PROOF OF THEOREM 2.1 FOR INTERIOR MANIFOLD POINTS

Suppose that $\{\vec{z_i}\}_{i=1}^k$ are sampled independently and uniformly at random from an n-dimensional ball. Let $\vec{w} = \frac{\sum_i \vec{z_i}}{\|\sum_i \vec{z_i}\|}$ and let S_{nk} be defined as in Equation (2). Define

$$X_i := \mathbf{1}_{\vec{w}\cdot\vec{z_i}\geq 0}\,,$$
$$Q_{nk} := \mathbb{P}[\vec{w}\cdot\vec{z_1}\geq 0]\,.$$

Proof of Theorem 2.1. By Chebyschev's inequality,

$$\mathbb{P}\Big[\big|S_{nk}/k - \frac{1}{2}\big| > \epsilon\Big] \leq \frac{\text{var}(S_{nk})}{k^2\epsilon^2} = \frac{Q_{nk}(1-Q_{nk})}{k\epsilon^2} + \frac{\text{cov}(X_1,X_2)}{\epsilon^2}\,.$$

The first summand is bounded above by $k^{-1}\epsilon^{-2}$ because $0 \leq Q_{nk} \leq 1$. By Lemma B.6, there are constants C_1, C_2 such that $|\text{cov}(X_1,X_2)| \leq C_1\sqrt{\frac{n}{k}} + C_2\frac{n}{k}$ for all $k \geq 3$. Therefore,

$$\mathbb{P}\Big[\big|S_{nk}/k - \frac{1}{2}\big| > \epsilon\Big] \leq \epsilon^{-2}\Big(C_1\sqrt{\frac{n}{k}} + (1+C_2)\frac{n}{k}\Big)\,.$$

□

Below, we prove Lemma B.6 and other supporting lemmas. First, we establish basic statistics about a point $\vec{z_1}$ drawn uniformly at random from an n-dimensional ball. The mean is $\vec{0}$ by symmetry of the ball.

Lemma B.1. *The covariance matrix for $\vec{z_1}$ is $\frac{1}{n+2}I_n$.*

Proof. First, we note that $\mathbb{E}[z_{1i}^2]$ is independent of i. Let $r = \|\vec{z_1}\|$ be the radial coordinate of $\vec{z_1}$. For any i, we have

$$n\mathbb{E}[z_{1i}^2] = \sum_{j=1}^n \mathbb{E}[z_{1j}^2] = \mathbb{E}[r^2] = \int_0^1 r^2\phi(r)dr$$

where $\phi(r)$ is the pdf for r in an n-dimensional ball. We have that $\phi(r)$ is proportional to the volume of an $(n-1)$-dimensional sphere (not ball) of radius r, so $\phi(r) = Ar^{n-1}$ for a constant A. Because $\int_0^1 \phi(r)dr = 1$, we must have $\phi(r) = nr^{n-1}$. Therefore,

$$\text{var}(z_{1i}) = \mathbb{E}[z_{1i}^2] = \frac{1}{n}\int_0^1 r^2\phi(r)dr = \int_0^1 r^{n+1}dr = \frac{1}{n+2}\,.$$

By symmetry of the ball,

$$\mathbb{E}[z_{1i}z_{1j}] = \mathbb{E}[-z_{1i}z_{1j}] = -\mathbb{E}[z_{1i}z_{1j}]$$

for $i \neq j$, so $\text{cov}(z_{1i}, z_{1j}) = \mathbb{E}[z_{1i}z_{1j}] = 0$. □

We will use the following lemma in an application of the Berry-Esseen theorem to bound the probability Q_{nk}.

Lemma B.2. *The third moment of z_{1i} is*

$$\mathbb{E}[|z_{1i}|^3] = \frac{4v_{n-1}}{v_n(n+3)(n+1)}\,,$$

where v_n is the volume of a unit n-dimensional ball.

Proof. The slice of the ball where $z_{1i} = t \in [-1,1]$ is an $(n-1)$-dimensional ball of radius $\sqrt{1-t^2}$, which has volume $v_{n-1}(1-t^2)^{\frac{n-1}{2}}$. Therefore, the pdf is

$$f(t) = A^{-1}v_{n-1}(1-t^2)^{\frac{n-1}{2}}\,,$$

where $A = \int_{-1}^{1} v_{n-1}(1-t^2)^{\frac{n-1}{2}} = v_n$. The third moment is

$$\mathbb{E}[|z_{1i}|^3] = 2\int_0^1 z_{1i}^3 f(t)dt = \frac{2v_{n-1}}{v_n}\int_0^1 t^3(1-t^2)^{\frac{n-1}{2}}dt\,.$$

Substituting $s = t^2$,

$$\mathbb{E}[|z_{1i}|^3] = \frac{v_{n-1}}{v_n}B\Big(2, \frac{n+1}{2}\Big) = \frac{v_{n-1}}{v_n}\frac{\Gamma(2)\Gamma(\frac{n+1}{2})}{\Gamma(2+\frac{n+1}{2})} = \frac{v_{n-1}}{v_n}\frac{\Gamma(\frac{n+1}{2})}{\frac{n+3}{2}\frac{n+1}{2}\Gamma(\frac{n+1}{2})} = \frac{4v_{n-1}}{v_n(n+3)(n+1)}\,.$$

□

Lemma B.3. *For all dimensions n and neighborhood sizes k,*

$$\frac{1}{2} \le Q_{nk} \le \frac{1}{2} + \frac{1}{\sqrt{2\pi}}\sqrt{\frac{n+2}{k-1}} + \frac{8C_{BE}}{\sqrt{2\pi(k-1)}}\,.$$

where C_{BE} is a universal constant from the Berry-Esseen theorem.

Proof. Note that

$$Q_{nk} = \mathbb{P}\left[\sum_{j>1}\vec{z_j}\cdot\vec{z_1} \ge -\left\|\vec{z_1}^2\right\|\right].$$

Therefore,

$$Q_{nk} \ge \mathbb{P}\left[\sum_{j>1}\vec{z_j}\cdot\vec{z_1} \ge 0\right] = \frac{1}{2}\,.$$

because $\|\vec{z_1}\|^2 \ge 0$. The last equality holds because $\sum_{j>1}\vec{z_j}$ and $\vec{z_1}$ are independent vectors in the n-ball that are equally likely to have angle $\theta \in [0, \pi/2]$ and angle $\pi - \theta \in [\pi/2, \pi]$.

Next, we prove the upper bound on Q_{nk}. Conditioning on $\vec{z_1} = \vec{u}$, we have

$$Q_{nk} = \mathbb{E}_{\vec{z_1}}\left[\mathbb{P}\Big[\sum_{j>1}\vec{z_j}\cdot\vec{u} \ge -\|\vec{u}\|^2 \mid \vec{z_1} = \vec{u}\Big]\right] \tag{4}$$

The sum $\sum_{j>1}\vec{z_j}\cdot\vec{u}$ is a sum of $k-1$ iid random variables with mean 0 because $\mathbb{E}[\vec{z_j}] = 0$. We will apply the Berry-Esseen theorem to approximate it by a normal random variable; as preliminaries, we need its variance and third moment of each term in the sum. By Lemma B.1, we have $\text{var}(\sum_{j>1}\vec{z_j}\cdot\vec{u}) = \frac{k-1}{n+2}\|\vec{u}\|^2$. To calculate the third moment $\mathbb{E}[|\vec{z_j}\cdot\vec{u}|^3]$, note that

$$\mathbb{E}\Big[|\vec{z_j}\cdot\vec{u}|^3\Big] = \mathbb{E}\Big[|\vec{z_j}\cdot\hat{u}|^3\Big]\,\|\vec{u}\|^3\,,$$

where $\hat{u} = \vec{u}/\|\vec{u}\|$. By rotational symmetry we may assume that $\hat{u} = (1, 0, \ldots, 0)$. Therefore, $\vec{z_j}\cdot\hat{u} = z_{j1}$ and so $\mathbb{E}[|\vec{z_j}\cdot\vec{u}|^3] = \mathbb{E}[|z_{j1}|^3]\|\vec{u}\|^3$. By Lemma B.2,

$$\mathbb{E}\Big[|\vec{z_j}\cdot\vec{u}|^3\Big] = \frac{4v_{n-1}}{v_n(n+1)(n+3)}\|\vec{u}\|^3 \le \frac{4v_{n-1}}{v_n(n+1)(n+3)}\,.$$

By the Berry-Esseen theorem,

$$\left|\mathbb{P}\Big[\sum_{j>1}\vec{z_j}\cdot\vec{u} \ge -\|\vec{u}\|^2\Big] - \Phi\Big(\sqrt{\frac{n+2}{k-1}}\|\vec{u}\|\Big)\right| \le \frac{4C_{\text{BE}}v_{n-1}}{v_n}\frac{(n+2)^{3/2}}{(n+1)(n+3)\sqrt{k-1}}$$

where C_{BE} is a universal constant. Finally, we upper bound $\Phi\Big(\sqrt{\frac{n+2}{k-1}}\|\vec{u}\|\Big)$. The cdf of a normal variable decreases monotonically after 0, so

$$\Phi\Big(\sqrt{\frac{n+2}{k-1}}\|\vec{u}\|\Big) \le \Phi\Big(\sqrt{\frac{n+2}{k-1}}\Big) = \frac{1}{2} + \int_0^{\sqrt{\frac{n+2}{k-1}}}\frac{1}{\sqrt{2\pi}}e^{-t^2/2}dt \le \frac{1}{2} + \frac{1}{\sqrt{2\pi}}\sqrt{\frac{n+2}{k-1}}\,.$$

All together,

$$\mathbb{P}\Big[\sum_{j>1} \vec{z_j} \cdot \vec{u} \geq -\|\vec{u}\|^2\Big] \leq \frac{1}{2} + \frac{1}{\sqrt{2\pi}}\sqrt{\frac{n+2}{k-1}} + \frac{4C_{\text{BE}}v_{n-1}}{v_n}\frac{(n+2)^{3/2}}{(n+1)(n+3)\sqrt{k-1}}$$

for all $\vec{u}$. Consequently, the right-hand side is an upper bound for Q_{nk} by Equation (4).

Finally, we upper bound v_{n-1}/v_n to finish the proof. The volume of a unit n-ball is $v_n = \pi^{n/2}/\Gamma(n/2+1)$. By Gautschi's inequality (Gautschi, 1959),

$$\frac{v_{n-1}}{v_n} = \frac{\Gamma(\frac{n}{2}+1)}{\Gamma(\frac{n}{2}+1)\sqrt{\pi}} \leq \sqrt{\frac{n+2}{2\pi}}\,.$$

Therefore,

$$\frac{4C_{\text{BE}}v_{n-1}}{v_n}\frac{(n+2)^{3/2}}{(n+1)(n+3)\sqrt{k-1}} \leq \frac{8C_{BE}}{\sqrt{2\pi(k-1)}}\,.$$

□

We will use Lemma B.4 and Lemma B.5 below to bound $\text{cov}(X_1, X_2)$ using an application of the multi-dimensional Berry-Esseen theorem.

Lemma B.4. *Suppose that $\hat{u_1}, \hat{u_2}$ are unit vectors in $\mathbb{R}^n$ with angle $\theta \neq 0, \pi$. Let U be the matrix whose columns are $\hat{u_1}, \hat{u_2}$ and let $R = \begin{bmatrix} 1 & \cos(\theta) \\ \cos(\theta) & 1 \end{bmatrix}$. If $\vec{z}$ is sampled uniformly at random from the n-ball, then*

$$\mathbb{E}\Big[\Big\|R^{-1/2}U^T\vec{z}\Big\|^3\Big] = \mathbb{E}\Big[\|P_U\vec{z}\|^3\Big] \leq \frac{2}{n+2}\,,$$

where P_U is the orthogonal projection matrix onto span$(\hat{u_1}, \hat{u_2})$.

Proof. First, note that $R = U^TU$. Therefore,

$$\begin{aligned}\Big\|R^{-1/2}U^T\vec{z}\Big\|^2 &= \vec{z}^TU(R^{-1/2})^TR^{-1/2}U^T\vec{z} \\ &= \vec{z}UR^{-1}U^T\vec{z} \\ &= \vec{z}U(U^TU)^{-1}U^T\vec{z} \\ &= \vec{z}P_U\vec{z} \\ &= \|P_U\vec{z}\|^2\,.\end{aligned}$$

Because $\|P_U\vec{z}\| \leq \|\vec{z}\| \leq 1$, we have

$$\mathbb{E}\Big[\Big\|R^{-1/2}U^T\vec{z}\Big\|^3\Big] = \mathbb{E}\Big[\|P_U\vec{z}\|^3\Big] \leq \mathbb{E}\Big[\|P_U\vec{z}\|^2\Big]\,. \tag{5}$$

Because P_U is an orthogonal projection matrix,

$$\mathbb{E}\Big[\|P_U\vec{z}\|^2\Big] = \mathbb{E}[\vec{z}^TP_U\vec{z}]\,.$$

We have

$$\mathbb{E}[\vec{z}^TP_U\vec{z}] = \mathbb{E}[\text{tr}(\vec{z}^TP_U\vec{z})] = \mathbb{E}[\text{tr}(P_U\vec{z}\vec{z}^T)] = \text{tr}\Big(P_U\mathbb{E}[\vec{z}\vec{z}^T]\Big)\,.$$

For $i \neq j$, we have $\mathbb{E}[(\vec{z}\vec{z}^T)_{ij}] = \mathbb{E}[\vec{z}_i]\mathbb{E}[\vec{z}_j] = 0$ because the components are independent and mean-centered. For $i = j$, we have $\mathbb{E}[z_i^2] = \frac{1}{n}\sum_r \mathbb{E}[z_r^2] = \frac{1}{n}\|\vec{z}\|^2$. Therefore, $\mathbb{E}[\vec{z}\vec{z}^T] = \frac{1}{n}\mathbb{E}[\|z\|^2]I_n$. By Lemma B.6, we have $\mathbb{E}[\|z\|^2] = \frac{n}{n+2}$. All together,

$$\mathbb{E}\Big[\|P_U\vec{z}\|^2\Big] = \frac{1}{n+2}\text{tr}(P_U)\,.$$

Because P_U is an orthogonal projection matrix onto a plane (the span of non-parallel vector $\hat{u_1}, \hat{u_2}$), its eigenvalues are $1, 1, 0, \ldots, 0$, so $\text{tr}(P_U) = 2$. By Equation (5), $\mathbb{E}\Big[\Big\|R^{-1/2}U^T\vec{z}\Big\|^3\Big] \leq \frac{2}{n+2}$. □

Lemma B.5. *Let* $R = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ *for some* $-1 \le \rho \le 1$*, and suppose that* $\vec{B} \sim N(0, R)$*. Define*

$$f(\tau_1, \tau_2, \rho) = \mathbb{P}[B_1 \le \tau_1, B_2 \le \tau_2] - \Phi(\tau_1)\Phi(\tau_2).$$

Then

$$|f(\tau_1, \tau_2, \rho) + f(\tau_1, \tau_2, -\rho)| \le \frac{4}{\sqrt{2\pi}}(|\tau_1| + |\tau_2|)\,.$$

Proof. We have $f(\tau_1, \tau_2, -\rho) = \mathbb{P}[B_1 \le \tau_1, -B_2 \le \tau_1] - \Phi(\tau_1)\Phi(\tau_2)$ because $B_1, -B_2$ are normally distributed with variance 1 and correlation $-\rho$. Therefore,

$$\begin{aligned} f(0,0,\rho) + f(0,0,-\rho) &= \mathbb{P}[B_1 \le 0, B_2 \le 0] + \mathbb{P}[B_1 \le 0, -B_2 \le 0] - 2\Phi(0)\Phi(0) \\ &= \mathbb{P}[B_1 \le 0] - \frac{1}{2} \\ &= 0\,. \end{aligned}$$

Therefore, for nonzero τ_1, τ_2,

$$\begin{aligned} &|f(\tau_1, \tau_2, \rho) + f(\tau_1, \tau_2, -\rho)| \\ &\quad \le |f(\tau_1, \tau_2, \rho) - f(0,0,\rho)| + |f(\tau_1, \tau_2, \rho) - f(0,0,-\rho)| + |f(0,0,\rho) - f(0,0,-\rho)| \\ &\quad = |f(\tau_1, \tau_2, \rho) - f(0,0,\rho)| + |f(\tau_1, \tau_2, \rho) - f(0,0,-\rho)| \end{aligned} \tag{6}$$

Therefore, it suffices to show that $|\partial f/\partial \tau_i| \le \frac{2}{\sqrt{2\pi}}$. Let $g(\tau_1, \tau_2, \rho) = \mathbb{P}[B_1 \le \tau_1, B_2 \le \tau_2]$, where ρ is the correlation of B_1, B_2. We will calculate $\partial f/\partial \tau_i$ for $i = 1$; the case $i = 2$ is symmetric. We have

$$\partial f/\partial \tau_1 = \frac{\partial g}{\partial \tau_1} - \varphi(\tau_1)\Phi(\tau_2)\,.$$

Letting $\varphi_\rho(b_1, b_2)$ denote the joint pdf for B_1, B_2, we have

$$\frac{\partial g}{\partial \tau_1} = \frac{\partial g}{\partial \tau_1}\int_{-\infty}^{\tau_1}\int_{-\infty}^{\tau_2} \varphi_\rho(b_1, b_2) db_2 db_1 = \int_{-\infty}^{\tau_2} \varphi_\rho(\tau_1, b_2) db_2 = \varphi(\tau_1)\mathbb{P}[B_2 \le \tau_2 \mid B_1 = \tau_1]\,.$$

To calculate the conditional CDF, we note that φ_ρ factors as

$$\begin{aligned} \varphi_\rho(b_1, b_2) &= \frac{1}{2\pi\sqrt{1-\rho^2}} \exp\Big(-\frac{1}{2(1-\rho^2)}[b_1^2 + b_2^2 - 2\rho b_1 z - 2]\Big) \\ &= \frac{1}{2\pi\sqrt{1-\rho^2}} \exp\Big(-\frac{1}{2(1-\rho^2)}[(1-\rho^2)b_1^2 + (b_2 - \rho b_1)^2]\Big) \\ &= \frac{1}{\sqrt{2\pi}} \exp\Big(-\frac{b_1^2}{2}\Big) \frac{1}{\sqrt{2\pi(1-\rho^2)}} \exp\Big(-\frac{1}{2(1-\rho^2)}(b_2 - \rho b_1)^2\Big) \\ &= \varphi(b_1) \frac{1}{\sqrt{2\pi(1-\rho^2)}} \exp\Big(-\frac{1}{2(1-\rho^2)}(b_2 - \rho b_1)^2\Big)\,, \end{aligned}$$

so the pdf for $B_2 \mid B_1 = \tau_1$ is the pdf of a normal random variable with mean ρb_1 and variance $1 - \rho^2$. Therefore,

$$\mathbb{P}[B_2 \le \tau_2 \mid B_1 = \tau_1] = \Phi\Big(\frac{\tau_2 - \rho\tau_1}{\sqrt{1-\rho^2}}\Big)\,.$$

Because $0 \le \Phi(x) \le 1$ for any x,

$$|\partial f/\partial \tau_1| = \varphi(\tau_1)\Big|\Phi\Big(\frac{\tau_2 - \rho\tau_1}{\sqrt{1-\rho^2}}\Big) - \Phi(\tau_2)\Big| \le \varphi(\tau_1) = \frac{1}{\sqrt{2\pi}} e^{-\tau_1^2/2} \le \frac{1}{\sqrt{2\pi}}\,. \tag{7}$$

The same upper bound holds for $|\partial f/\partial \tau_2|$. Combining Equation (6) and Equation (7) completes the proof. □

Lemma B.6. *We have* $var(X_i) = Q_{nk}(1 - Q_{nk})$ *for all i, and there are constants* C_1, C_2 *such that*

$$cov(X_1, X_2) \le C_1\sqrt{\frac{n}{k}} + C_2\frac{n}{k}$$

for all $k \ge 3$*.*

Proof. For the variance, observe that X_i is a Boolean random variable with probability of success Q_{nk}. For the covariance, we have

$$\text{cov}(X_1, X_2) = \mathbb{E}[X_1 X_2] - Q_{nk}^2 = \mathbb{P}[X_1 = 1 \text{ and } X_2 = 1] - Q_{nk}^2 \,. \tag{8}$$

By Lemma B.3, there are constants A_1, A_2 such that

$$\left| Q_{nk}^2 - \frac{1}{4} \right| \leq A_1 \sqrt{\frac{n}{k}} + A_2 \frac{n}{k} \,. \tag{9}$$

for all $k \geq 2$.

The focus of the rest of the proof is to upper bound $|\mathbb{P}[X_1 = 1 \text{ and } X_2 = 1] - \frac{1}{4}|$. Let $\hat{v} = \sum_{j>2} \vec{z_j}$. For $i = 1, 2$, we have

$$\sum_j \vec{z_j} \cdot \vec{z_i} = \hat{v} \cdot \vec{z_i} + \|\vec{z_i}\|^2 + \vec{z_1} \cdot \vec{z_2} \,,$$

so $X_i = 1$ if and only if $\hat{v} \cdot \vec{z_i} \geq -\|\vec{z_i}\|^2 - \vec{z_1} \cdot \vec{z_2}$. Conditioning on $\vec{z_1} = \vec{u_1}$ and $\vec{z_2} = \vec{u_2}$, we get

$$\mathbb{P}[X_1 = 1 \text{ and } X_2 = 1] = \mathbb{E}_{\vec{z_1}, \vec{z_2}}[\mathbb{P}[X_1 = 1 \text{ and } X_2 = 1 \mid \vec{z_1} = \vec{u_1},\, \vec{z_2} = \vec{u_2}]] \,,$$

where the conditional probability is

$$\mathbb{P}[X_1 = 1 \text{ and } X_2 = 1 \mid \vec{z_1} = \vec{u_1},\, \vec{z_2} = \vec{u_2}] = \mathbb{P}[\hat{v} \cdot \vec{u_1} \geq -a_1 \text{ and } \hat{v} \cdot \vec{u_2} \geq -a_2] \,, \tag{10}$$

where $a_i = \|\vec{u_i}\|^2 - \vec{u_1} \cdot \vec{u_2}$ for $i = 1, 2$.

Let $Y_i = \hat{v} \cdot \vec{u_i}$ for $i = 1, 2$, and consider the 2D vector $\vec{Y} = (Y_1, Y_2)$. The right-hand side of Equation (10) is $\mathbb{P}[\vec{Y} \geq (-a_1, -a_2)]$. Therefore,

$$\mathbb{P}[X_1 = 1 \text{ and } X_2 = 1] = \mathbb{E}_{\vec{z_1}, \vec{z_2}}[\mathbb{P}[\vec{Y} \geq (-a_1, -a_2)]]$$

We have that $\vec{Y}$ is the sum of $k - 2$ iid 2D random variables:

$$\vec{Y} = \sum_{j>2} (\vec{z_j} \cdot \vec{u_1}, \vec{z_j} \cdot \vec{u_2})$$

with mean 0 because $\mathbb{E}[\vec{z_j}] = 0$ by symmetry of the ball. Using the fact that the variance of the ith component of $\vec{z_j}$ is $\frac{1}{n+2}$ by Lemma B.1, we get that the covariance matrix for $\vec{Y}$ is

$$C_{\vec{Y}} = \frac{k-2}{n+2} \begin{bmatrix} \|u_1\|^2 & \vec{u_1} \cdot \vec{u_2} \\ \vec{u_1} \cdot \vec{u_2} & \|u_2\|^2 \end{bmatrix} .$$

We normalize $\vec{Y}$ so that each component has variance 1. Let $\hat{Y}_i = Y_i / \sigma_{Y_i}$ and let $\tau_i = a_i / \sigma_{Y_i}$. The main probability of interest is

$$\mathbb{P}[\vec{Y} \geq (-a_1, -a_2)] = \mathbb{P}[\hat{Y} \geq (-\tau_1, -\tau_2)] \,,$$

where

$$|\tau_i| = \sqrt{\frac{n+2}{k-2}} \left| \frac{\left\| \vec{u_i}^2 \right\| - \vec{u_1} \cdot \vec{u_2}}{\|\vec{u_i}\|} \right| \leq 2\sqrt{\frac{n+2}{k-2}} \tag{11}$$

because $\vec{u_1}$ and $\vec{u_2}$ are in the unit ball. The covariance matrix for $\hat{Y}$ is

$$R := \begin{bmatrix} 1 & \cos(\theta) \\ \cos(\theta) & 1 \end{bmatrix} ,$$

where θ is the angle between $\vec{u_1}$ and $\vec{u_2}$.

Let $\vec{B} \sim N(\vec{0}, R)$. We will upper bound $|\mathbb{P}[\hat{Y} \geq (-\tau_1, -\tau_2)] - \mathbb{P}[\vec{B} \geq (-\tau_1, -\tau_2)]|$ using the multi-dimensional Berry-Esseen theorem. We can write $\hat{Y}$ as the sum of $(k-2)$ iid random variables $\hat{Y} = \sum_{j=3}^{k} \vec{w_j}$, where

$$\vec{w_j} := (\vec{z_j} \cdot \vec{u_1} / \sigma_{Y_1}, \vec{z_j} \cdot \vec{u_2} / \sigma_{Y_2}) = \sqrt{\frac{n+2}{k-2}} (\vec{z_j} \cdot \hat{u_1}, \vec{z_j} \cdot \hat{u_2}) = \sqrt{\frac{n+2}{k-2}} U^T \vec{z}$$

where $\hat{u_i} = \vec{u_i}/\|u_i\|$ and U is the $(n \times 2)$ matrix with columns $\hat{u_1}, \hat{u_2}$. The covariance matrix R is invertible whenever $\theta \neq 0, \pi$. Therefore when $\theta \neq 0, \pi$, the multi-dimensional Berry-Esseen theorem implies that

$$|\mathbb{P}[\hat{Y} \geq (-\tau_1, -\tau_2)] - \mathbb{P}[\vec{B} \geq (-\tau_1, -\tau_2)]| \leq C_{BE} 2^{1/4} \gamma_{k-2}\,,$$

where C_{BE} is a universal constant and

$$\gamma_{k-2} = \sum_{j=3}^{k} \mathbb{E}\left[\left\|R^{-1/2}\vec{w_j}\right\|_2^3\right] = (k-2)\mathbb{E}\left[\left\|R^{-1/2}\vec{w_3}\right\|_2^3\right] = \frac{(n+2)^{3/2}}{\sqrt{k-2}}\mathbb{E}\left[\left\|R^{-1/2}U^T\vec{z}_3\right\|_2^3\right].$$

By Lemma B.4, we have $\gamma_{k-2} \leq \sqrt{\frac{n+2}{k-2}}$ when $\theta \neq 0, \pi$. Therefore,

$$|\mathbb{P}[\hat{Y} \geq (-\tau_1, -\tau_2)] - \mathbb{P}[\vec{B} \geq (-\tau_1, -\tau_2)]| \leq C_{BE} 2^{1/4} \sqrt{\frac{n+2}{k-2}}$$

for all fixed $\vec{u_1}, \vec{u_2}$ such that $\theta \neq 0, \pi$. Vectors $\vec{z_1}, \vec{z_2}$ have angle $\theta \neq 0, \pi$ with probability 1, so

$$\left|\mathbb{E}_{\vec{z}_1,\vec{z}_2}\left[\mathbb{P}[\hat{Y} \geq (-\tau_1, -\tau_2)] - \mathbb{P}[\vec{B} \geq (-\tau_1, -\tau_2)]\right]\right| \leq C_{BE} 2^{1/4} \sqrt{\frac{n+2}{k-2}}\,. \tag{12}$$

Next, we show that $\Phi(\tau_1)\Phi(\tau_2)$ is a good approximation to $\mathbb{P}[\vec{B} \geq (-\tau_1, -\tau_2)]$ in the expectation over $\vec{z_1}, \vec{z_2}$. First, note that

$$\mathbb{P}[\vec{B} \geq (-\tau_1, -\tau_2)] = \mathbb{P}[\vec{B} \leq (\tau_1, \tau_2)]\,.$$

Use the notation $\mathbb{P}_\theta[\vec{B} \leq (\tau_1, \tau_2)]$ to denote the probability when the angle between $\vec{u_1}$ and $\vec{u_2}$ is θ. By Lemma B.5,

$$|\mathbb{P}_\theta[\vec{B} \leq (\tau_1, \tau_2)] + \mathbb{P}_{\pi-\theta}[\vec{B} \leq (\tau_1, \tau_2)] - 2\Phi(\tau_1)\Phi(\tau_2)| \leq \frac{4}{\sqrt{2\pi}}(|\tau_1| + |\tau_2|)\,.$$

Therefore when we integrate over all $\vec{u_1}, \vec{u_2}$ in the n-ball, we get

$$\left|\mathbb{E}_{\vec{z}_1,\vec{z}_2}\left[\mathbb{P}_\theta[\vec{B} \leq (\tau_1, \tau_2)] - \Phi(\tau_1)\Phi(\tau_2)\right]\right| \leq \frac{2}{\sqrt{2\pi}}(|\tau_1| + |\tau_2|)\,.$$

By Equation (11),

$$\left|\mathbb{E}_{\vec{z}_1,\vec{z}_2}\left[\mathbb{P}_\theta[\vec{B} \leq (\tau_1, \tau_2)] - \Phi(\tau_1)\Phi(\tau_2)\right]\right| \leq \frac{8}{\sqrt{2\pi}}\sqrt{\frac{n+2}{k-2}} \tag{13}$$

Additionally, we note that

$$\frac{1}{2} \leq \Phi(\tau_i) \leq \phi\Big(\sqrt{\frac{n+2}{k-2}}\Big) \leq \frac{1}{2} + \int_0^{\sqrt{\frac{n+2}{k-2}}} \frac{1}{\sqrt{2\pi}} e^{-t^2/2} dt \leq \frac{1}{2} + \frac{1}{\sqrt{2\pi}}\sqrt{\frac{n+2}{k-2}}\,.$$

Consequently,

$$\left|\Phi(\tau_1)\Phi(\tau_2) - \frac{1}{4}\right| \leq \frac{1}{\sqrt{2\pi}}\sqrt{\frac{n+2}{k-2}} + \frac{1}{2\pi}\Big(\frac{n+2}{k-2}\Big)\,. \tag{14}$$

By Equation (8), Equation (9), Equation (12), Equation (13), and Equation (14), there are constants C_1, C_2 such that

$$\begin{aligned}
|\mathrm{cov}(X_1, X_2)| &\leq \left|\mathbb{E}_{\vec{z}_1,\vec{z}_2}\left[\mathbb{P}[\hat{Y} \geq (-\tau_1, -\tau_2)] - \mathbb{P}[\vec{B} \geq (-\tau_1, -\tau_2)]\right]\right| \\
&\quad + \left|\mathbb{E}_{\vec{z}_1,\vec{z}_2}\left[\mathbb{P}[\vec{B} \geq (-\tau_1, -\tau_2)] - \Phi(\tau_1)\Phi(\tau_2)\right]\right| \\
&\quad + \left|\Phi(\tau_1)\Phi(\tau_2) - \frac{1}{4}\right| + \left|Q_{nk}^2 - \frac{1}{4}\right| \\
&\leq C_1\sqrt{\frac{n}{k}} + C_2\frac{n}{k}
\end{aligned}$$

for all $k \geq 3$.

□

B.2 Proof of Theorem 2.2 for boundary points

Next, we consider the case where $\{\vec{z_i}\}_{i=1}^k$ is a set of points sampled from an n-dimensional half ball, modeling the boundary score of a point x on the boundary of an n-dimensional manifold.

Proof of Theorem 2.2. The random variable $1 - \frac{S_{nk}}{k}$ is non-negative. By Markov's inequality,

$$\mathbb{P}\Big[1 - \frac{S_{nk}}{k} \geq \epsilon\Big] \leq \frac{\mathbb{E}\big[1 - \frac{S_{nk}}{k}\big]}{\epsilon} = \frac{1 - \mathbb{P}[\vec{w} \cdot \vec{z_1} \geq 0]}{\epsilon} = \frac{\mathbb{P}[\vec{w} \cdot \vec{z_1} < 0]}{\epsilon}\,.$$

Applying Lemma B.11 completes the proof. □

Below, we prove the supporting lemmas, starting with statistics about a point $\vec{z_1}$ sampled uniformly at random from the half ball. Without loss of generality, we assume that it is the half ball where the first coordinate is ≥ 0.

Notation B.7. Let $\vec{\mu}^{(n)} = \mathbb{E}[\vec{z_1}]$ and let $\sigma_i^{(n)}$ denote the standard deviation of the ith component z_{1i}. Let $F^{(n)}(t)$ denote the cdf for the first coordinate z_{11}, and let $f^{(n)}(t)$ denote the pdf. We will denote the volume of a unit n-dimensional ball by v_n.

Lemma B.8. *The pdf $f^{(n)}(t)$ for the first coordinate of $\vec{z_1}$ is*

$$f^{(n)}(t) = \frac{2v_{n-1}}{v_n}(1-t^2)^{\frac{n-1}{2}}\,.$$

Proof. The slice of the half ball where $z_{11} = t$ is an $(n-1)$-dimensional ball of radius $\sqrt{1-t^2}$, which has volume $v_{n-1}(1-t^2)^{\frac{n-1}{2}}$. Therefore,

$$f^{(n)}(t) = A^{-1}v_{n-1}(1-t^2)^{\frac{n-1}{2}}\,,$$

where $A = \int_0^1 v_{n-1}(1-t^2)^{\frac{n-1}{2}}dt$. This is the volume $v_n/2$ of the half ball. □

Lemma B.9. *The mean $\vec{\mu}^{(n)}$ has components*

$$\mu_i^{(n)} = \begin{cases} \frac{2v_{n-1}}{v_n(n+1)}\,, & i = 1\,, \\ 0\,, & i > 1\,. \end{cases}$$

Proof. By symmetry, $\mu_i^{(n)} = 0$ for $i > 1$. For $i = 1$,

$$\mu_1^{(n)} = 2\frac{v_{n-1}}{v_n}\int_0^1 t(1-t^2)^{\frac{n-1}{2}}$$

by Lemma B.8. Using the substitution $s = 1 - t^2$,

$$\mu_1^{(n)} = \frac{v_{n-1}}{v_n}\int_0^1 s^{\frac{n-1}{2}}ds = \frac{2v_{n-1}}{v_n(n+1)}\,.$$

□

Lemma B.10. *The variance $(\sigma_i^{(n)})^2$ of the ith component of $\vec{z_1}$ is*

$$(\sigma_i^{(n)})^2 = \begin{cases} \frac{1}{n+2} - (\mu_1^{(n)})^2\,, & i = 1\,, \\ \frac{1}{n+2}\,, & i > 1\,. \end{cases}$$

Proof. The same proof as Lemma B.1 shows that $\mathbb{E}[z_{1i}^2] = \frac{1}{n+2}$ for all i, because the pdf $\phi(r)$ for $\|\vec{z_1}\|$ is the same in both the full and half ball.

Therefore, the variance is

$$(\sigma_i^{(n)})^2 = \mathbb{E}[z_{1i}^2] - (\mu_i^{(n)})^2 = \begin{cases} \frac{1}{n+2} - (\mu_1^{(n)})^2\,, & i = 1\,, \\ \frac{1}{n+2}\,, & i > 1\,. \end{cases}$$

□

Lemma B.11. *Let $\{\vec{z_i}\}_{i=1}^k$ be points drawn independently and uniformly at random from an n-dimensional half ball where the first coordinate is ≥ 0, and let $\vec{w} = \frac{\sum_i \vec{z_i}}{\|\sum_i \vec{z_i}\|}$. Then*

$$\mathbb{P}[\vec{w} \cdot \vec{z_1} < 0] \leq \frac{3}{\sqrt[3]{4}} \frac{n+1}{\sqrt[3]{k}} .$$

Proof. Define $\vec{v} = \frac{1}{k} \sum_j \vec{z_j}$. Note that $\vec{w} \cdot \vec{z_1} < 0$ if and only if $\vec{v} \cdot \vec{z_1} < 0$, so $\mathbb{P}[\vec{w} \cdot \vec{z_1} < 0] = \mathbb{P}[\vec{v} \cdot \vec{z_1} < 0]$. First, will show that $\vec{v}$ concentrates near $\vec{\mu}^{(n)}$. Then, we will condition on $\vec{v}$ being near $\vec{\mu}^{(n)}$ to derive an upper bound on $\mathbb{P}[\vec{v} \cdot \vec{z_1} < 0]$. When $\vec{v}$ is sufficiently close to $\vec{\mu}^{(n)}$, the conditional probability is small.

First, we bound the L^∞ norm of $\vec{v} - \vec{\mu}^{(n)}$. Let $\delta > 0$. For the ℓth component,

$$\mathbb{P}\big[|v_\ell - \mu_\ell^{(n)}| > \delta|\big] = \frac{\frac{1}{k}(\sigma_\ell^{(n)})^2}{\delta^2} \leq \frac{1}{k(n+2)\delta^2}$$

by Chebyschev's inequality and Lemma B.10. Let A_δ be the event that $\left\|\vec{v} - \vec{\mu}^{(n)}\right\|_\infty \leq \delta$. Applying a union bound yields

$$\mathbb{P}[A_\delta^c] \leq \sum_{\ell=1}^n \mathbb{P}\big[|v_\ell - \mu_\ell^{(n)}| > \delta|\big] \leq \frac{n}{k(n+2)\delta^2} .$$

Now we condition on the event A_δ. For any $\delta > 0$, we have that

$$\begin{aligned} \mathbb{P}[\vec{v} \cdot \vec{z_1} < 0] &= \mathbb{P}[\vec{v} \cdot \vec{z_1} < 0 \mid A_\delta]\mathbb{P}[A_\delta] + \mathbb{P}[\vec{v} \cdot \vec{z_1} < 0 \mid A_\delta^c]\mathbb{P}[A_\delta^c] \\ &\leq \mathbb{P}[\vec{v} \cdot \vec{z_1} < 0 \mid A_\delta] + \mathbb{P}[A_\delta^c] \\ &\leq \mathbb{P}[\vec{v} \cdot \vec{z_1} < 0 \mid A_\delta] + \frac{n}{k(n+2)\delta^2} . \end{aligned} \tag{15}$$

We can write

$$\vec{v} \cdot \vec{z_1} = \sum_{\ell=1}^n \mu_\ell^{(n)} z_{1\ell} + \sum_{\ell=1}^n (v_\ell - \mu_\ell^{(n)}) z_{1\ell} = \mu_1^{(n)} z_{i1} + \sum_{\ell=1}^n (v_\ell - \mu_\ell^{(n)}) z_{1\ell} ,$$

where the last equality follows by Lemma B.9. If $\left\|\vec{v} - \vec{\mu}^{(n)}\right\|_\infty \leq \delta$, then

$$\Big|\sum_{\ell=1}^n (v_\ell - \mu_\ell^{(n)}) z_{1\ell}\Big| \leq \sum_\ell |v_\ell - \mu_\ell^{(n)}| \cdot |z_{1\ell}| \leq \delta \|\vec{z_1}\|_1 \leq \delta\sqrt{n}\|\vec{z_1}\|_2 \leq \delta\sqrt{n} .$$

Therefore,

$$\vec{v} \cdot \vec{z_1} \geq \mu_1^{(n)} z_{i1} - \delta\sqrt{n}$$

when $\left\|\vec{v} - \vec{\mu}^{(n)}\right\|_\infty \leq \delta$. This yields

$$\mathbb{P}[\vec{v} \cdot \vec{z_1} < 0 \mid A_\delta] \leq \mathbb{P}[\mu_1^{(n)} z_{i1} \leq \delta\sqrt{n}] = F^{(n)}\Big(\frac{\delta\sqrt{n}}{\mu_1^{(n)}}\Big) .$$

Combining with Equation (15) gives us

$$\mathbb{P}[\vec{v} \cdot \vec{z_1} < 0] \leq F^{(n)}\Big(\frac{\delta\sqrt{n}}{\mu_1^{(n)}}\Big) + \frac{n}{k(n+2)\delta^2} , \tag{16}$$

where $F^{(n)}$ is the CDF for the first coordinate of $\vec{z_1}$. The CDF is monotonically increasing and satisfies

$$F(t) = \int_0^t f^{(n)}(s)ds \leq t \max_{s \in [0,1]} f^{(n)}(s) .$$

By Lemma B.8, the pdf $f^{(n)}(s)$ is monotonically decreasing, so

$$\max_{s\in[0,1]} f^{(n)}(s) = f^{(n)}(0) = \frac{2v_{n-1}}{v_n} = (n+1)\mu_1^{(n)}\,.$$

Therefore,

$$F^{(n)}\Big(\frac{\delta\sqrt{n}}{\mu_1^{(n)}}\Big) \le \sqrt{n}(n+1)\delta\,.$$

Substituting into Equation (16) gives us

$$\mathbb{P}[\vec{v}\cdot\vec{z_1} < 0] \le \sqrt{n}(n+1)\delta + \frac{n}{k(n+2)\delta^2} =: g(\delta) \tag{17}$$

for all δ.

To finish the proof, we will optimize the choice of δ as a function of n and k. The right-hand side $g(\delta)$ approaches ∞ as $\delta \to 0$ or $\delta \to \infty$, so it achieves a minimum. Write $g(\delta) = a\delta + b\delta^{-2}$, where $a = \sqrt{n}(n+1)$ and $b = \frac{n}{k(n+2)}$. The only critical point occurs when

$$\delta_* = \sqrt[3]{\frac{2b}{a}}\,.$$

At the critical point, we have

$$g(\delta_*) = \frac{3}{\sqrt[3]{4}}\sqrt[3]{a^2 b} = \frac{3}{\sqrt[3]{4}}\sqrt[3]{\frac{n^2(n+1)^2}{(n+2)k}} \le \frac{3}{\sqrt[3]{4}}\sqrt[3]{\frac{(n+1)^4}{(n+1)k}} = \frac{3}{\sqrt[3]{4}}\,\frac{n+1}{\sqrt[3]{k}}\,.$$

By Equation (17),

$$\mathbb{P}[\vec{v}\cdot\vec{z_1} < 0] \le \frac{3}{\sqrt[3]{4}}\,\frac{n+1}{\sqrt[3]{k}}\,.$$

□

C Methodological details for experiments

C.1 Intrinsic dimension estimation

HP requires local dimension estimation (Step 3 of Algorithm 1). Unless otherwise stated, we use a spectral gap threshold τ to determine the dimension. Given a neighborhood $\{x_{ij}\}_j$ for point x_i, let $\lambda_1 \ge \lambda_2 \ge \cdots \ge \lambda_D$ be the eigenvalues of the covariance matrix (for the local neighborhood) in descending order. The local dimension estimate at x_i is the minimum index m such that $\lambda_m/\lambda_{m+1} > \tau$. If no such index exists, we use dimension D. The default threshold is $\tau = 7$.

C.2 AUC values for synthetic data experiments

In Section 4, we calculated AUC values for interior vs. boundary points of disks (and n-dimensional balls) embedded in $\mathbb{R}^D$. Because the ground-truth boundary of a disk or ball has measure 0, there are not any points on the ground-truth disk (or n-ball) boundary, so we defined the "empirical" boundary to be the set of points with norm larger than a threshold τ_B and the "empirical" interior to be the set of points with norm smaller than a threshold τ_I. We used $\tau_B = .95$ and $\tau_I = .8$ in Section 4. For the disks embedded in $\mathbb{R}^D$, the norm was calculated with respect to the first two coordinates (the plane that contains the disk), and for the n-dimensional balls in $\mathbb{R}^n$, we used the n-dimensional Euclidean norm. In this section, we present the AUC values for different choices of $\tau_B \in \{.91, \ldots, .98\}$. Our conclusions do not qualitatively change when we use a different threshold τ_B.

In Figure 5, we considered the AUC values for 2D disks embedded in $\mathbb{R}^D$ with increasing ambient dimension D, at fixed boundary threshold $\tau_B = .95$. In Figure 20, we show the AUC values for varying thresholds $\tau_B \in \{.91, \ldots, .98\}$. At each threshold τ_B, the DU method performs best, followed by HP, followed by LoDD.

In Figure 7(B), we calculated AUC values for 2D disks embedded in $\mathbb{R}^3$ with increasing noise $\sigma \in [.01, .05]$, at fixed boundary threshold $\tau_B = .95$. In Figure 21, we show the AUC values for

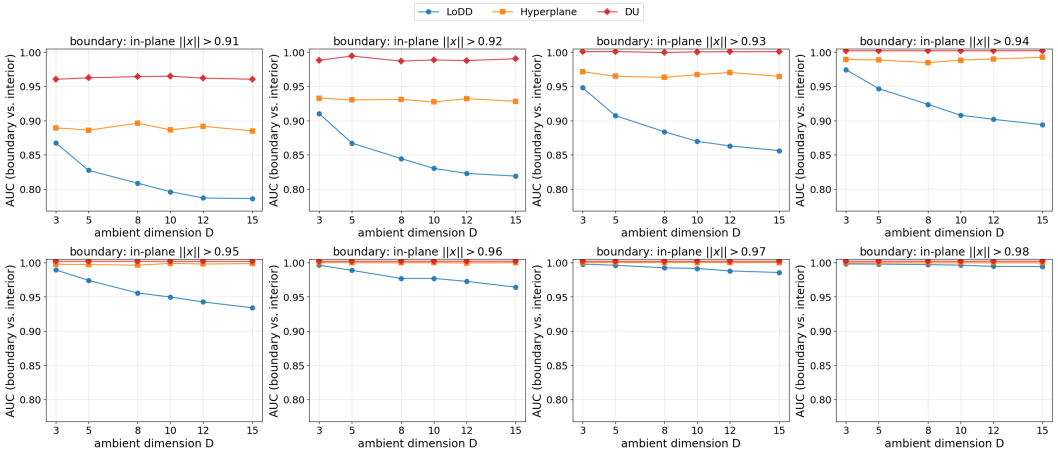


Figure 20: Point clouds are sampled from 2D disks embedded in $\mathbb{R}^D$ for $D = 3, 5, 8, 10, 12, 15$ with Gaussian noise level $\sigma = .01$. Each point cloud has $N = 10,000$ points, and we use neighborhood size $k = 100$. Interior points are defined to be points with norm less than .8, and boundary points are defined to be points with norm larger than τ_B, which increases from $\tau_B = .91$ to $\tau_B = .98$ across the panels. In each panel, we plot the AUC values as a function of ambient dimension D for each of the three methods.

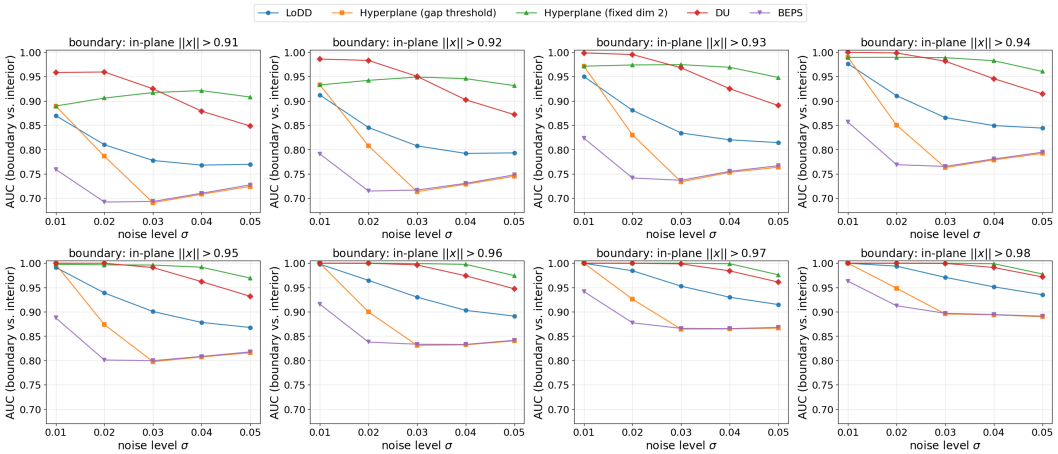


Figure 21: At each noise level σ, we sample a point cloud from a 2D disk in $\mathbb{R}^3$ with isotropic Gaussian noise. Each point cloud has $N = 10,000$ points, and we use neighborhood size $k = 100$. Interior points are defined to be points with norm less than .8, and boundary points are defined to be points with norm larger than τ_B, which increases from $\tau_B = .91$ to $\tau_B = .98$ across the panels. In each panel, we plot the AUC values as a function of noise level σ for each of the three methods.

varying thresholds $\tau_B \in \{.91, \ldots, .98\}$. For $\tau_B \geq .95$, the relative performance of each method is the same as when $\tau = .95$, where the hyperplane method with fixed dimension does best, closely followed by DU. For $.91 \leq \tau_B \leq .94$, DU outperforms every method at the lowest noise levels $\sigma = .01$ and $\sigma = .02$.

In Figure 7(C), we calculated AUC values for 2D disks embedded in $\mathbb{R}^D$ with increasing dimension D and fixed noise level $\sigma = .05$, at fixed boundary threshold $\tau_B = .95$. In Figure 22, we show the AUC values for varying thresholds $\tau_B \in \{.91, \ldots, .98\}$. At all threshold τ_B, the hyperplane method with fixed dimension performs best, followed by DU, followed for LoDD, followed by BEPS and the hyperplane method with a spectral gap threshold condition.

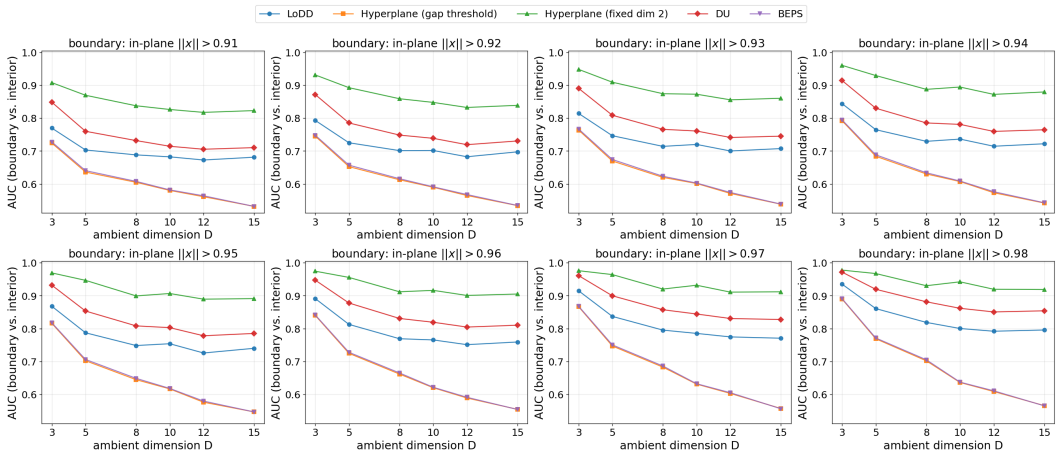


Figure 22: At each ambient dimension D, we sample a point cloud from a 2D disk in $\mathbb{R}^D$ with isotropic Gaussian noise $\sigma = .05$. Each point cloud has $N = 10,000$ points, and we use neighborhood size $k = 100$. Interior points are defined to be points with norm less than .8, and boundary points are defined to be points with norm larger than τ_B, which increases from $\tau_B = .91$ to $\tau_B = .98$ across the panels. In each panel, we plot the AUC values as a function of ambient dimension D for each of the four methods.

C.3 Spatial Transcriptomics

C.3.1 Data preprocessing

The raw data has 741 spatial spots and $27,984$ genes.

Quality control. We do gene quality control by discarding genes that are detected at fewer than 3 spots. This results in the removal of $12,372$ genes.

Library-size normalization. We normalize each spot to a common total count of $10,000$. More precisely, if x_{ig} is the number of transcripts of gene g at spot i, the library size at spot i is $L_i = \sum_g x_{ig}$. We rescale the counts $x_{ig} := (x_{ig}/L_i) \times 10,000$ so that each spot has library size $10,000$.

Log transform. For each spot i and gene g, we apply a log transform $x_{ig} := \log(1 + x_{ig})$, where x_{ig} is the output from the library-size normalization step.

Highly variable genes. We select the $3,000$ most highly-variable genes, discarding the rest.

Scaling. We normalize each gene so that it has mean 0 and variance 1 by mapping each x_{ig} to its z-score. More precisely, if μ_g and σ_g are the mean and standard deviation for gene g expression across spots, we calculate $x_{ig} := (x_{ig} - \mu_g)/\sigma_g$. We clip values at max z-score 10, i.e., $x_{ig} := \min(x_{ig}, 10)$, so that no single gene dominates the PCA step below.

PCA dimension reduction. Let $x_i = (x_{ig})_g$ be the gene expression values at spot i after the preprocessing steps above. We use PCA to reduce $\{x_i\}_i$ to 30 dimensions.

C.3.2 Neighborhood calculation for the boundary detection algorithms

Both HP and DU require a local neighborhood at each point (see lines 2 of Algorithm 1 and Algorithm 2, respectively). We define the neighborhoods as follows.

For each spot, we concatenate the 30-dimensional PCA gene expression coordinates with the 2D spatial coordinates, yielding a 32-dimensional vector at each spot. The gene and spatial coordinates have very different scales, so to determine the nearest neighbors of each spot, it does not make sense to naively calculate Euclidean distance in $\mathbb{R}^{32}$.

Instead, we consider spatial and gene distance independently. For integer k', let $N^g(k', i)$ denote the nearest k' neighbors of the ith spot with respect to gene coordinates (Euclidean distance in $\mathbb{R}^{30}$), and let $N^s(k', i)$ denote the nearest k' neighbors of the ith spot with respect to spatial coordinates (Euclidean distance in $\mathbb{R}^2$). At neighborhood size parameter k, we define the neighborhood of spot i to be the intersection

$$N^g(k', i) \cap N^s(k', i)$$

for the minimum integer k' such that the intersection has size k.

C.4 LLM Activation Geometry

In Section 6, we applied our boundary detection algorithms to study LLM activation geometry. The results for Pythia-2.8B were shown in Figure 10, and the results for Llama-3.1-8B were shown in Figure 19. Here, we detail our methodology.

C.4.1 Data set

We stream documents from the C4 (English) data set (Raffel et al., 2020), keeping those that have at least 400 characters excluding whitespace. For each document and each $k \in \{15, 20, \ldots, 150\}$, we remove all newlines and then re-insert them every $\leq k$ characters, breaking at the nearest word boundary $\leq k$ so that each line has at most k characters.[5] We continue extracting documents from C4 until we reach at least $250,000$ non-newline tokens in total. This is satisfied by the first 28 documents. (See Table 2 for total token counts for Pythia-2.8B and Llama-3.1-8B.) This process yields 28 documents $\times 28$ widths $= 784$ total sequences. We cap each sequence at 512 tokens, including newlines.

Each non-newline token is labeled by its character position i and line width $w \in \{15, 20, \ldots, 150\}$. The character position of a token is defined to be the number of characters used from the beginning of the line through the end of the token.

	Pythia-2.8B	Llama-3.1-8B
Total tokens	271,100	269,651
Non-newline tokens	250,437	250,641

Table 2: Number of tokens extracted for each model.

C.4.2 Models and layer selection

We test two models, Pythia-2.8B and Llama-3.1-8B, which have dimensions $D = 2560$ and $D = 4096$, respectively. Both models have 32 layers. For each token in our data set, we extract the residual-stream activations at layer 4 in both models. We chose layer 4 after visual inspection of the averaged character-counting manifold for candidate layers 4, 11, 16, 24, and 29.

C.4.3 Averaged manifold

For each (i, w), we average the residual-stream vectors at layer 4 for tokens with character position i and line width w. We discard (i, w) if there are fewer than 5 tokens with position i and line width w. This gives us $N = 2,311$ points for the Pythia-2.8B manifold and $N = 2,335$ points for the Llama-3.1-8B, manifold.[6]

C.4.4 Boundary detection on the averaged manifold

First, we use PCA to reduce the averaged manifold ($N \approx 2300$ for each model) to 30 dimensions. Then, we run the DU method and the hyperplane method with fixed dimension 2, both with neigh-

[5] Occasionally, we encounter a token with $\geq k$ characters itself. These tokens are put onto their own line, thus creating a line that is over k characters. These lines are rare.

[6] If every token had position $1 \leq i \leq w$, there would be at most $2,310$ possible labels (i, w). For a line of width w, every token has position $1 \leq i \leq w$ with the exception of tokens that have length $\geq w$. A token with more than w characters forms its own line. This gives us occasional tokens with label (i, w) for $i > w$, which is why the averaged manifolds for both Pythia-2.8B and Llama-3.1-8B have more than $2,310$ points.

borhood size $k = 100$. (The manifold is intrinsically 2-dimensional because it is parameterized by character position i and line width 2.)

The hyperplane method yields a vector $\vec{w}$, in local PCA coordinates, at each point, which we lift to a vector $\vec{w}_{\text{amb}}$ in the ambient 30-dimensional PCA basis. The DU method gives us a vector $\vec{v_i} \in \mathbb{R}^{30}$ at each point. In Figure 10 and Figure 19, we show the vectors for points whose boundary scores are in the top decile.

C.4.5 Boundary detection on the un-averaged, per-token manifold

Instead of averaging the residual-stream activations for each (i, w), we consider the entire point cloud of size size $N \approx 250,000$ (one point per token). Again, we use PCA to reduce to 30 dimensions. Then, we run the DU method and the hyperplane method with neighborhood size $k = 100$. We use a spectral gap threshold condition (with default threshold $\tau = 7$) for the hyperplane method instead of fixing dimension 2 because the intrinsic dimension is no longer 2. At each (i, w), we average the boundary scores for all tokens at position i within a line of width w.

C.4.6 3D visualization

For both models, the first principal component of the averaged manifold is dominated by a few outliers, so in Figure 10 and Figure 19, we plot in PC2-PC3-PC4 coordinates.

D Other boundary detection variants

Both CO (Wang et al., 2018) and BEPS (Li & Maguire, 2011) use the vector $\vec{v_i} := \frac{1}{k}\sum_j \frac{x_{ij}-x_i}{\|x_{ij}-x_i\|}$ at each point x_i, where $\{x_{ij}\}$ are its k nearest neighbors. The CO score at point x_i is

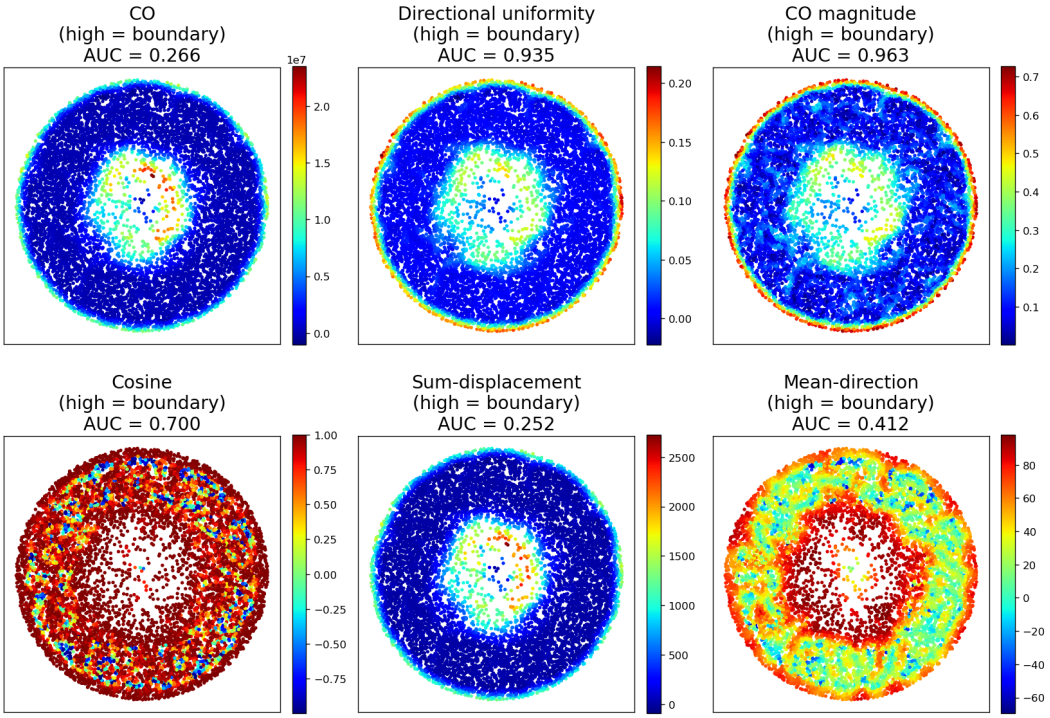
$$k^2 \sum_{j=1}^{k} m_i^{-1} m_j^{-1} \vec{v_i} \cdot \vec{v_j}\,,$$

where $m_i^{-1} = \sum_j \|x_{ij} - x_i\|$. The CO score is high on low-density points, regardless of whether they are interior or boundary points, so the CO method struggles with non-uniform density. The difference between our DU method and the CO method is that we removed the term $m_i^{-1} m_j^{-1}$ and normalized the score so that it is always less than or equal to 1.

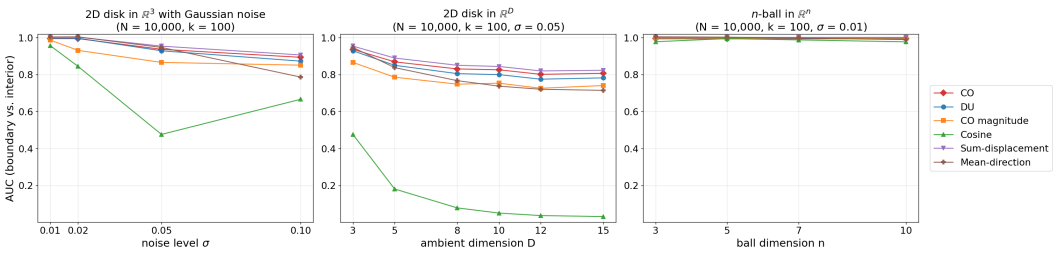
We tested several other boundary detection variants (see Table 3), with results shown in Figure 23. Each variant uses either the vector $\vec{v_i}$ or a slightly different vector $\vec{u_i} := \sum_j (x_{ij} - x_i)$ where the neighbor-displacement vectors are not normalized, and then each variant has a score that is constructed from the chosen vector. The cosine variant uses $\vec{v_i}$ but measures the cosine between $\vec{v_i}$ and $\sum_j \vec{v_j}$ instead of taking the dot product; this variant isolates the effect of neighbor alignment between vectors, which was the rationale in Wang et al. (2018). We find that the cosine variant works poorly because the vectors of *all* neighbors tend to be similar due to the fact that their neighborhoods are similar, regardless of whether they are on a boundary or not. The magnitude variant $\|\vec{v_i}\|$ works almost as well as DU on disks and n-balls, but as discussed in Section 3, the DU variant is more robust to noise. The two variants that use $\vec{u_i}$ do not work well with non-uniform density because $\|\vec{u_i}\|$ is large when the density is low. The DU variant was the best overall performer across varying noise levels, varying dimension, and non-uniform density.

Variant	Vector	Score
CO (Wang et al., 2018)	$\vec{v_i}$	$k^2 \sum_{j=1}^{k} m_i^{-1} m_j^{-1} \vec{v_i} \cdot \vec{v_j}$
Directional uniformity	$\vec{v_i}$	$\sum_{j=1}^{k} \vec{v_i} \cdot \vec{v_j}$
CO magnitude	$\vec{v_i}$	$\|\vec{v_i}\|$
Cosine	$\vec{v_i}$	$\cos(\vec{v_i}, \sum_j \vec{v_j})$
Sum displacement	$\vec{u_i}$	$\sum_j \vec{u_i} \cdot \vec{u_j}$
Mean direction	$\vec{u_i}$	$\sum_j \cos(\vec{u_i}, \vec{u_j})$

Table 3: Other boundary-detection methods that we tested.



(a)



(b)

Figure 23: We compare the variants from Table 3. (A) Scores on the non-uniformly sampled disk from Figure 12 with $N = 10,000$ points and neighborhood size $k = 100$. AUROC scores are for boundary points (norm $> .98$) vs deep interior points (norm $< .23$), like in Figure 12. AUROC scores for boundary vs. interior points on 2D disks and n-balls, where interior and boundary points are defined as having norms $< .8$ and $> .95$, respectively.