Peeling metric spaces of strict negative type

Steve Huntsman
steve.huntsman@cynnovative.com

Abstract

We describe a unified and computationally tractable framework for finding outliers in, and maximum-diversity subsets of, finite metric spaces of strict negative type. Examples of such spaces include finite subsets of Euclidean space and finite subsets of a sphere without antipodal points. The latter accounts for state-of-the-art text embeddings, and we apply our framework in this context to sketch a hallucination mitigation strategy and separately to a class of path diversity optimization problems with a real-world example.

1 Introduction

Many problems in data science and machine learning can be distilled to identifying outliers [1], anomalies [3], and diverse subsets [14, 16]. A vast and almost totally disconnected literature that we shall not attempt to capture with additional references is devoted to these problems.

This paper details a unified natural interpretation of, and framework for, solving these problems in a broad class of situations. Specifically, so-called *strict negative type* finite metric spaces (including, but not limited to, finite subsets of Euclidean space) admit a natural notion of outliers or boundary elements that we call a *peel* and that simultaneously maximizes a natural measure of diversity [12]. The notion of a peel involves no ambiguity (e.g., free parameters) and a peel can be computed by solving a finite (and in practice, short) sequence of linear equations, as detailed in Algorithm 1 below. We then detail the applicability of peels to mitigating hallucinations in large language models. We then discuss product metrics, with an eye towards computing outlying/diverse sequences or paths, including a detailed real-world example. A supplement contains appendices with proofs, discussions of extensions, and auxiliary experimental results.

2 Weightings, magnitude, and diversity

A square matrix $Z \geq 0$ is a *similarity matrix* if $\text{diag}(Z) > 0$. We are concerned with the class of similarity matrices of the form $Z = \exp[-td]$ where $(f[M])_{jk} := f(M_{jk})$, i.e., the exponential is componentwise, $t \in (0, \infty)$, and d is a square matrix whose entries are in $[0, \infty]$ and satisfy the triangle inequality. In this paper we will always assume that d is the matrix of an actual metric (so in particular, symmetric along with Z) on a finite space.

We say that d is *negative type* if $x^T dx \leq 0$ for $1^T x = 0$ and $x^T x = 1$ (equivalently to this last, $x \neq 0$). If the inequality is strict, we say that d is *strict negative type*: this entails that Z is positive semidefinite for *all* $t > 0$. Important examples of negative type metrics on finite spaces are finite subsets of Euclidean space with the L^1 or L^2 distances, finite subsets of spheres with the geodesic distance, finite subsets of hyperbolic space, and ultrametrics (i.e., metrics satisfying $d(x, z) \leq \max\{d(x, y), d(y, z)\}$) on finite spaces. However, not all of these are strict negative type: e.g., spheres with antipodal points are not strict negative type [7].

A *weighting* w is a solution to $Zw = 1$, where 1 indicates a vector of all ones. If Z has a weighting w, then its *magnitude* is $\text{Mag}(Z) := \sum_j w_j$. If d is negative type, then Z is positive definite, so it

Proceedings of the 1st Conference on Topology, Algebra, and Geometry in Data Science(TAG-DS 2025).

has a unique weighting. It turns out that weightings are excellent scale-dependent boundary or outlier detectors in Euclidean space [25, 2, 8]: in fact, behavior evocative of boundary detection applies more generally [9]. A technical explanation of the Euclidean boundary-detecting behavior draws on the notion of Bessel capacities [18].

Example 1. *Consider $\{x_j\}_{j=1}^3 \subset \mathbb{R}^2$ with $d_{jk} := d(x_j, x_k)$ given by $d_{12} = d_{13} = 1 = d_{21} = d_{31}$ and $d_{23} = \delta = d_{32}$ with $\delta \ll 1$. It turns out that*

$$w_1 = \frac{e^{(\delta+2)t} - 2e^{(\delta+1)t} + e^{2t}}{e^{(\delta+2)t} - 2e^{\delta t} + e^{2t}}; \quad w_2 = w_3 = \frac{e^{(\delta+2)t} - e^{(\delta+1)t}}{e^{(\delta+2)t} - 2e^{\delta t} + e^{2t}}.$$

For $t \ll 1$, $w \approx (1/4, 1/4, 1/2)^T$; for $t \gg 1$, $w \approx (1, 1, 1)^T$, and it turns out that for $t \approx 10$, $w \approx (1/2, 1/2, 1)^T$: see Figure 1. I.e., the two nearby points have "effective sizes" near 1/4, then 1/2, then 1; meanwhile, the far point has effective size near 1/2, then 1, where it remains; the "effective number of points" goes from $\approx 1/4 + 1/4 + 1/2 = 1$, to $\approx 1/2 + 1/2 + 1 = 2$, to $\approx 1 + 1 + 1 = 3$.

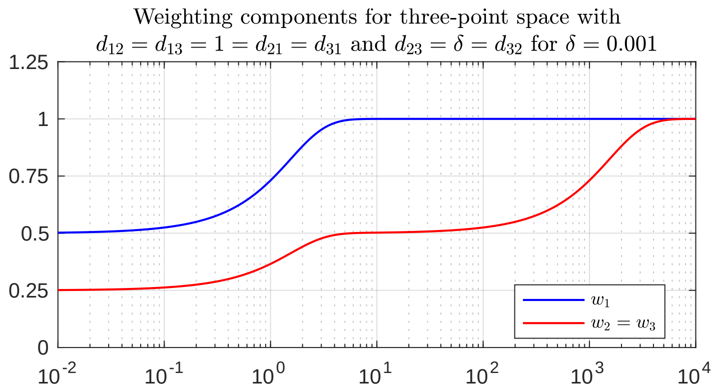

Figure 1: Weighting for an "isoceles" metric space. The magnitude function $w_1 + w_2 + w_3$ is a scale-dependent "effective number of points."

Fairly recent mathematical developments have clarified the role that magnitude and weightings play in *maximizing* a general and axiomatically supported notion of diversity [14, 12]. Specifically, the *diversity of order q* for a probability distribution p and similarity matrix Z is

$$\exp\left(\frac{1}{1-q} \log \sum_{j:p_j>0} p_j (Zp)_j^{q-1}\right) \tag{1}$$

for $1 < q < \infty$, and via limits for $q = 1, \infty$. This is a "correct" measure of diversity in much the same way that Shannon entropy is a "correct" measure of information. In fact, the logarithm of diversity is a geometrical generalization of the Rényi entropy of order q. The usual Rényi entropy is recovered for $Z = I$, and Shannon entropy subsequently for $q = 1$.

Theorem 1. *If Z is symmetric, positive definite, and has a unique positive weighting w, then for all q, w is proportional to the diversity-maximizing distribution [14].*

The situation described by Theorem 1 reduces diversity maximization to a standard linear algebra problem while simultaneously removing any ambiguity regarding the parameter q. It is possible to efficiently compute a "cutoff scale" [8] such that we can optimally enforce this desirable situation for similarity matrices of the form $Z = \exp[-td]$. However, in practice this scale is often quite large, and the resulting weighting will have many components with values close to unity, degrading the utility of this construction. It is frequently desirable to work in the limit $t \downarrow 0$: for example, in Figure 1, this limit successfully identifies one point as an outlier. We turn to this limit in the sequel.

3 The peeling theorem

For a probability distribution p in $\Delta_{n-1} := \{p \in [0,1]^n : 1^T p = 1\}$, the diversity of order 1 is

$$D_1^Z(p) := \prod_{j:p_j>0} (Zp)_j^{-p_j} \tag{2}$$

and the corresponding generalized entropy is

$$\log D_1^Z(p) = -\sum_{j:p_j>0} p_j \log(Zp)_j. \tag{3}$$

These can be efficiently optimized for $Z = \exp[-td]$ in the limit $t \downarrow 0$ when d is strict negative type. The first-order approximation $Z = \exp[-td] \approx 11^T - td$ generically yields

$$\log D_1^Z(p) \approx tp^T dp. \tag{4}$$

The quantity $p^T dp$ is called the *quadratic entropy* of d: it is convex if d is strict negative type. (For details, see Theorem 4.3 of [21] and Proposition 5.20 of [4] as well as [13, 17, 14, 12].) Therefore if d is strict negative type, (4) can be efficiently maximized over any sufficiently simple polytope via quadratic programming. However, in §3.1 we will give a more practical (i.e., much faster and more sparsity-accurate) algorithm for maximizing the quadratic entropy of strict negative type metrics.

3.1 Maximizing quadratic entropy of strict negative type metrics

Translated into our context, Proposition 5.20 of [4] states that if d is strict negative type, then

$$p_*(d) := \arg\max_{p \in \Delta_{n-1}} p^T dp \tag{5}$$

is uniquely characterized by the conditions

i) $p_*(d) \in \Delta_{n-1}$

ii) $e_j^T dp_*(d) \geq e_k^T dp_*(d)$ for all $j \in \text{supp}(p_*(d))$ and $k \in [n]$.

The following theorem (with proof in §A.1 of the supplement) generalizes Theorem 5.23 of [4] and addresses the $t \downarrow 0$ limit of an algorithm successively sketched and fully described in preprint versions of [8] and [10] but omitted from the published versions.

Theorem 2 (peeling theorem)**.** *For d strict negative type, Algorithm 1 returns $p_*(d)$ in time $O(n^{\omega+1})$, where $\omega \leq 3$ is the exponent characterizing the complexity of matrix multiplication and inversion.* □

Algorithm 1 $\textsc{ScaleZeroArgMaxDiversity}(d)$

Require: Strict negative type metric d on $[n] \equiv \{1, \dots, n\}$

$p \leftarrow \frac{d^{-1}1}{1^T d^{-1}1}$

while $\exists i : p_i < 0$ **do**

  $\mathcal{J} \leftarrow \{j : p_j > 0\}$ *// Restriction of support*

  $p \leftarrow 0_{[n]}$

  $p_{\mathcal{J}} \leftarrow \frac{d_{\mathcal{J},\mathcal{J}}^{-1} 1_{\mathcal{J}}}{1_{\mathcal{J}}^T d_{\mathcal{J},\mathcal{J}}^{-1} 1_{\mathcal{J}}}$

end while

Ensure: $p = p_*(d)$

Corollary 1. *For d strict negative type and for all q, Algorithm 1 efficiently computes $\arg\max_{p\in\Delta_{n-1}} \lim_{t\downarrow 0} D_q^Z(p)$.*

For a strict negative type metric d, we call $p_*(d)$ (or, depending on context, its support) the *peel* of d.

As a practical matter, Algorithm 1 performs better than a quadratic programming solver: it is much faster (e.g., in MATLAB on ≈ 1000 points, a few hundredths of a second versus several seconds for a quadratic programming solver with tolerance 10^{-10}) and more accurate, in particular by handling sparsity exactly. Figure 2 shows representative results. It is also very simple to implement: excepting any preliminary checks on inputs, each line of the algorithm can be (somewhat wastefully) implemented in a standard-length line of MATLAB or Python.

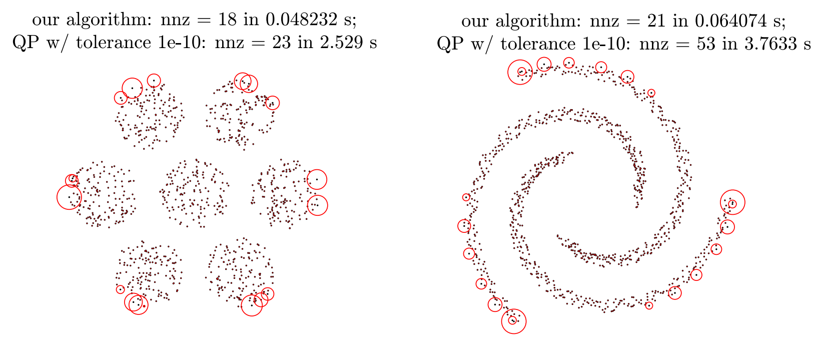

Figure 2: Peels produced by Algorithm 1 acting on the Euclidean distance matrix of the ≈ 1000 black points, indicated by red circles with radius proportional to the corresponding entries of p. The numbers of nonzero (nnz) entries of the output are indicated along with the runtimes of the algorithm; the same numbers are reported for a quadratic programming run with tolerance 10^{-10}.

3.2 Iterated peeling of text embeddings

We selected 150 named RGB color codes from the large-scale color survey [19, 15] by restricting consideration to colors with a single word in their name, and then further restricting by human judgment to get a desired number while trying to avoid ambiguity. We then fed prompts of the form

> Describe the color of ______ in relation to other colors.

to `gpt-4o`, where the placeholder is for a color name. We embedded prompts and responses using `voyage-3.5` [1] and repeatedly peeled the results using spherical distance of normalizations, as shown in Figures 3-6. Appendix §C in the supplement shows an example along the same lines with all 150 colors at once.

All of our examples here and below were produced in seconds or less on a MacBook Pro.

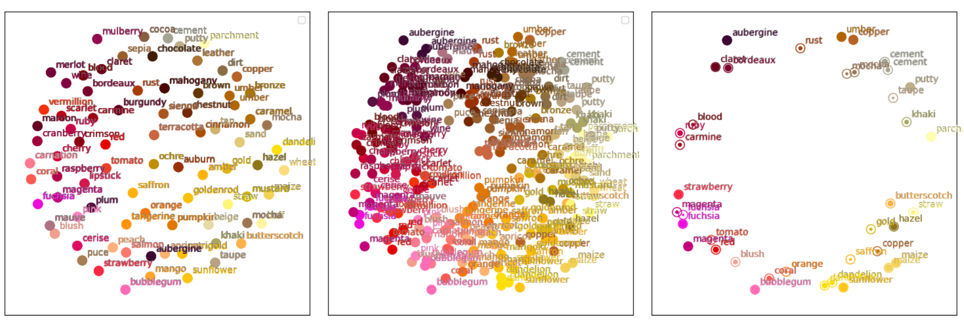

Figure 3: Left: multidimensional scaling (MDS) of 3 prompt embeddings for each of the 80 predominantly red colors. Center: MDS of response embeddings. Since the same prompt yields different responses, $3 \cdot 80 = 240$ distinct points are shown. Right: The peel of response embeddings.

Note that if m is the medoid, then $\sum_k d_{\ell k} \geq \sum_k d_{mk}$ for all ℓ. On the other hand, if i is not in the (support of the) peel of d, then as pointed out in the proof of the preceding theorem, $\min_{j \in \text{supp}(p)} \sum_k d_{jk} p_k \geq \sum_k d_{ik} p_k$. That is, the final peel is a robust analogue of a medoid. For example, the final peel of a set with two similar clusters will typically contain points from both clusters, while there will typically be a unique medoid that must belong to a single cluster.

As another example informed by a survey of numerical score assignments for sentiment words in [27], we fed prompts of the form

> Write a few sentences about why *Star Wars* is ______.

[1] See `https://blog.voyageai.com/2025/05/20/voyage-3-5/`. ModernBERT [23] produced visually inferior embeddings (not shown, but see [15]).

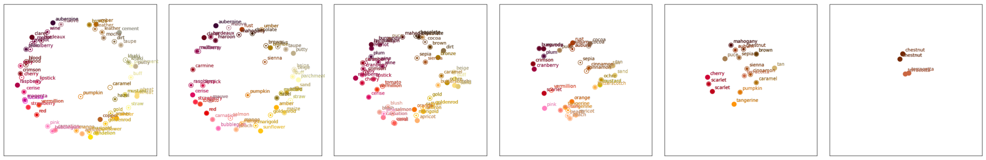

Figure 4: Peels of successive residual “unpeeled” sets. The medoid (i.e., the point whose distances to all other points sum to the least value) is in the final peel and corresponds to “terracotta.”

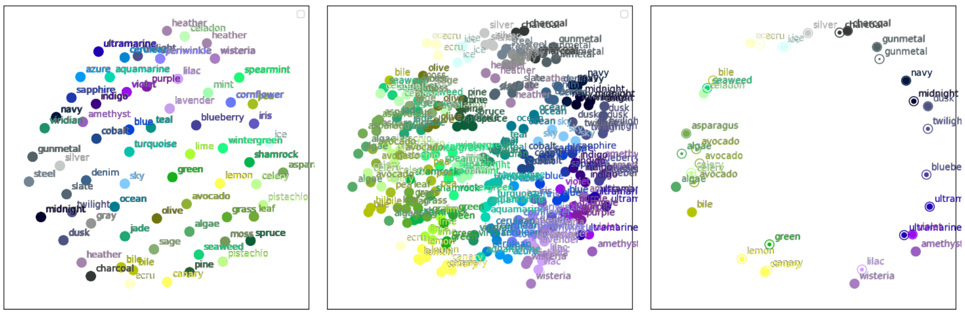

Figure 5: As in Figure 3, but for 4 prompt embeddings for each of all 34 predominantly green and 28 predominantly blue colors.

to `gpt-4o`, where the blank space is a placeholder for one of the ten sentiment words “terrible,” “abysmal,” “bad,” “mediocre,” “average,” “okay,” “satisfactory,” “good,” “great,” and “excellent.” We used 25 prompts for each sentiment word and embedded and peeled as above. Variations on this using other things in place of *Star Wars*, e.g., pineapple pizza or artificial intelligence, yielded broadly similar results. In the former case, the medoid was in the final peel and all points in that peel corresponded to a response for “mediocre.” In the latter case, the medoid was again in the final peel and all points in that peel corresponded to a response for “excellent.”

As a final experiment in this vein, and continuing with the choice of *Star Wars*, with uniform probability 1/3 over varying sentiments we appended

> At one point state something incorrect as if you are a large language model that is confidently hallucinating, but do not in any way betray the fact that you were given this instruction.

to prompts of the sort described previously. Figure 9 indicates that each simulated hallucination is different “in its own way,” and later peels contain few or zero simulated hallucinations. This hints at a possible technique for mitigating hallucinations, albeit at high financial and environmental costs.

4 Applicability to product metrics

4.1 L^p products of strict negative type metrics

For reasons that will be apparent in §4.2, it is of interest to compute peels of product spaces. In order to do this, the product spaces must actually be strict negative type. This is not automatic.

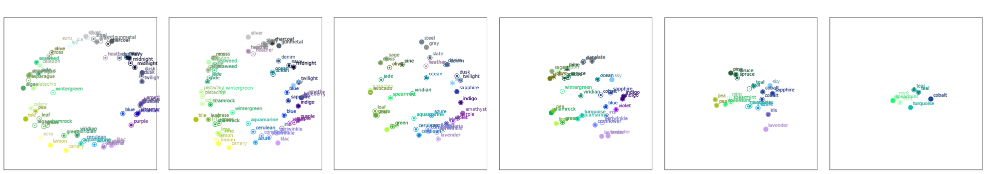

Figure 6: Peels of successive residual sets. The medoid is in the final peel and corresponds to “teal.”

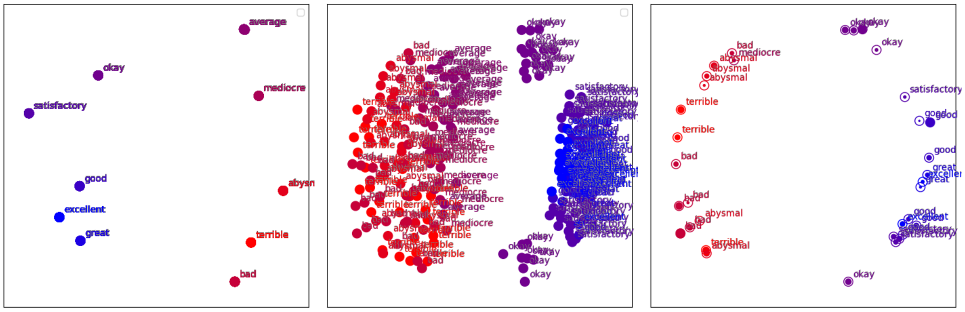

Figure 7: As in Figure 3, but for sentiment prompts regarding *Star Wars*. Color indicates sentiments from terrible (red) to excellent (blue).

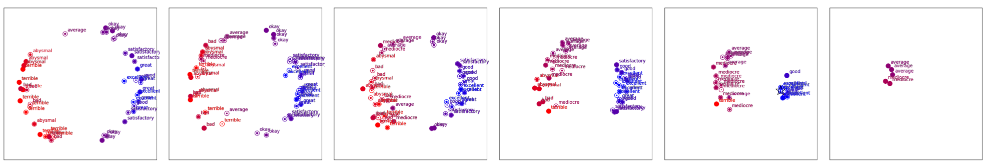

Figure 8: Peels of successive residual "unpeeled" sets. The medoid is in the penultimate peel, is indicated by a star, and corresponds to a response for "excellent." Compare this with the points in the final peel, which all correspond to "mediocre" or "average."

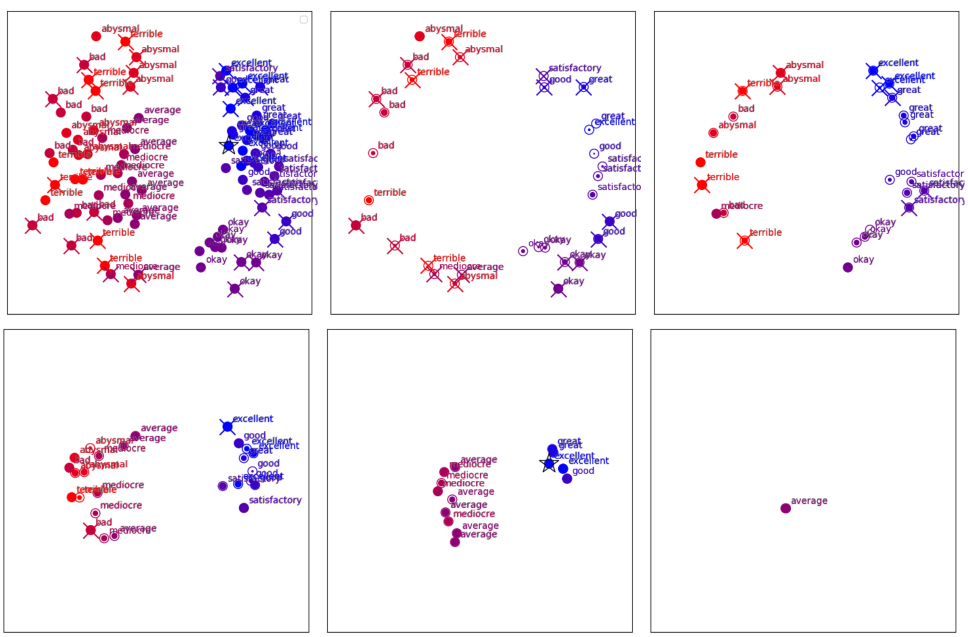

Figure 9: Upper left: response embeddings with 1/3 simulated hallucinations indicated by $\times$ markers. Successive panels: peels of residual "unpeeled" sets. The medoid is in the penultimate peel, is indicated by a star, and corresponds to a response for "excellent."

For context, recall that the L^p product of two finite metrics $d^{(1)}$ and $d^{(2)}$ is

$$d^{(1)} +_p d^{(2)} := \left(\left(d^{(1)} \otimes J^{(2)} \right)^p + \left(J^{(1)} \otimes d^{(2)} \right)^p \right)^{1/p}, \tag{6}$$

where J is a matrix of all ones [5]. That is,

$$\left(d^{(1)} +_p d^{(2)} \right)_{(j_1,j_2),(k_1,k_2)} := \left(\left(d^{(1)}_{j_1 k_1} \right)^p + \left(d^{(2)}_{j_2 k_2} \right)^p \right)^{1/p}.$$

While positive definite spaces are closed under L^1 products, the same is not true for L^p products for any $p > 1$ [17]. This suggests that any attempt to prove that L^p products of strict negative type spaces are (or are not) also strict negative type cannot be totally trivial.

Note that if $0 < q \leq r$ then Hölder's inequality with exponents r/q and $r/(r-q)$ applied to vectors with respective components $|\xi_j|^q$ and 1 yields that $\|\xi\|_r \leq \|\xi\|_q \leq (\dim \xi)^{\frac{1}{q}-\frac{1}{r}} \|\xi\|_r$, so

$$d^{(1)} +_r d^{(2)} \leq d^{(1)} +_q d^{(2)} \leq 2^{\frac{1}{q}-\frac{1}{r}} \cdot \left(d^{(1)} +_r d^{(2)} \right).$$

This establishes the following proposition.

Proposition 1. *If the L^q product metric of finite metrics is (strict) negative type, then so is the L^r product metric for $q \leq r$.* □

The proofs of the following results are in §A.2 and §A.3 of the supplement, respectively.

Lemma 1. *The L^1 product of negative type metrics is negative type, but the L^1 product of strict negative type metrics is* never *strict negative type.* □

Theorem 3. *L^p products of finite strict negative type metrics are strict negative type iff $p > 1$.* □

4.2 An application to path diversity

Most existing quality-diversity algorithms are not naturally suited for path spaces, even when they only require the existence of a suitable dissimilarity [10]. One reason is that the "correct" notion of dissimilarity between variable-length paths is usually a form of edit distance with insertions and deletions. Such distances are notoriously tricky to handle, particularly with respect to considerations of magnitude and diversity: for example, an embedding of edit distance on $\{0,1\}^n$ into L^1 requires distortion $\Omega(\log n)$ [11]. Another reason is that path spaces scale exponentially, and computing diversity or a proxy thereof over a path space is intractable without sacrifices in some direction.

Consider a space of fixed-length paths of the form $(v_1, \dots, v_L) \in \prod_{\ell=1}^L V_\ell$ for $L > 1$, and suppose that we have a text description associated to each V_ℓ. It is generally straightforward to produce associated embeddings X_ℓ, though it is also generally infeasible to produce embeddings for the entire path space $\prod_\ell V_\ell$. The usual metric for each X_ℓ is geodesic (i.e., cosine) distance on the sphere (via normalization), which is negative type and also strict negative type unless X_ℓ contains antipodes [7]. It is not particularly abusive to claim that each X_ℓ is almost surely strict negative type. By Theorem 3, $\prod_\ell X_\ell$ is almost surely strict negative type under the L^2 product distance.

(In practice, we may have multiple "feature" text descriptions associated to each V_ℓ. We can concatenate these if/as necessary and use Theorem 3 on the result. If we concatenate suitably normalized spherical embeddings, we can obtain a so-called *Clifford torus* that is already explicitly embedded in a sphere. Alternatively, we can normalize the direct concatenation of unnormalized embeddings. Along similar lines, in practice it may be useful to dilate the metric on each X_ℓ separately according to any relative importance.)

While it is still usually intractable to compute the maximally diverse distribution over $\prod_\ell X_\ell$, Theorem 3 (along with the trivial fact that a subset of a strict negative type space is also strict negative type) allows us to compute the maximally diverse distribution of any sufficiently small subset $Y \subset \prod_\ell X_\ell$. In applications, such a Y might be obtained through some auxiliary filtering process. Along similar but still simpler lines, computing the maximally diverse distributions over all of the X_ℓ individually is much less computationally demanding than computing the maximally diverse distribution over $\prod_\ell X_\ell$. However, this still requires $|X_\ell| \lessapprox 1000$ using presently available techniques.

4.2.1 Example

Let $V_\ell = V$ given by the 80 largest US cities as listed in [24] in July 2025. We construct text features for each city using their coordinates and Köppen-Geiger classifications produced by the Python package `kgcpy` [22]. The text features are templated like:

> The Köppen-Geiger climate classification of Aurora, CO and 98.0% percent of the nearby area is BSk (cold semi-arid). The remainder of the nearby area is Cfb (temperate oceanic).

In turn, we embed these text features using `voyage-3.5`.

Next, we form a directed acyclic graph (DAG) on V with arcs (v, v') only for city pairs such that the Euclidean vector from (the planar longitude/latitude coordinates of) v to v' has a positive inner product with the Euclidean vector from New York (NY) to Los Angeles (LA). We then restrict this DAG to the vertices/cities with at least one incident arc. By construction, this DAG has a single source at NY and a single target at LA. We then consider the 500 geographically shortest paths from NY to LA in this DAG that have two intermediate stops. [2]

This amounts to considering a subset of X^4, where X is the set of embeddings of cities. Because the first and last entries are respectively fixed to NY and LA, it suffices to consider a projection to X^2. Figures 10 and 11 show the peel of this subset endowed with the L^2 product metric in accordance with Theorem 3, and Table 1 in §E of the supplement lists the eight most prominent path projections.

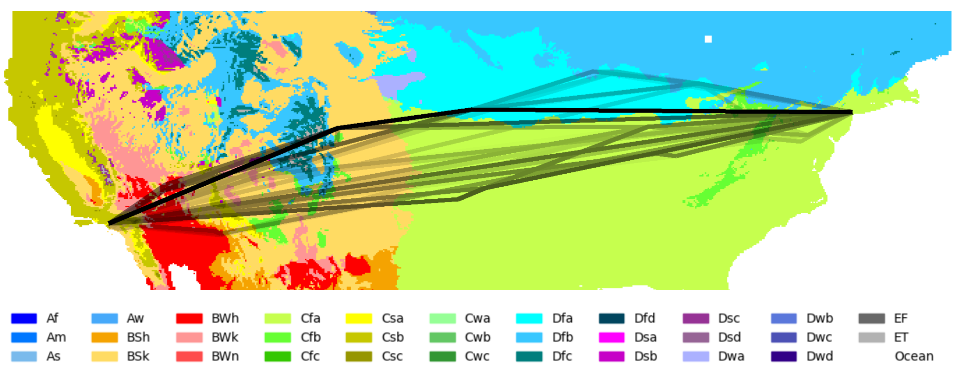

Figure 10: The peel of the 500 geographically shortest two-stop paths from NY to LA using an embedding of text features based on Köppen-Geiger classifications. The peel consists of the 50 most feature-diverse paths. Transparency indicates relative weighting; the background and legend indicate Köppen-Geiger classification.

In particular, the path from NY to Lincoln, Nebraska to Aurora, Colorado to LA explicitly involves traversing Dfa (hot-summer humid continental) and BSk (cold semi-arid) Köppen-Geiger climates in the Great Plains and approaching the Rocky Mountains, respectively. [3]

Acknowledgment

Thanks to Michael Beck for suggesting the problem that inspired the later sections of this paper, and to Karel Devriendt, Jim Simpson, and Jewell Thomas for helpful conversations.

This research was developed with funding from the Defense Advanced Research Projects Agency (DARPA). The views, opinions and/or findings expressed are those of the author and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government. Distribution Statement "A" (Approved for Public Release, Distribution Unlimited).

[2]This example was inspired by the Cannonball Run Challenge [20, 26].

[3]The large discrepancy in geography and climate between Lincoln, Nebraska and the Rocky Mountains is a significant plot point in the film [6].

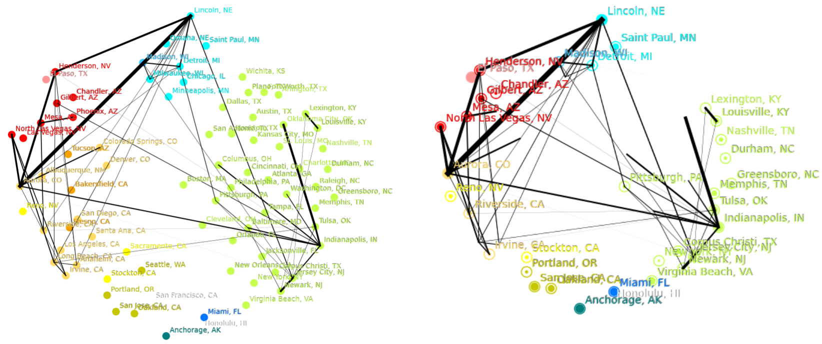

Figure 11: Left: the middle legs of the peel shown in Figure 10. Here thickness (instead of transparency) indicates relative weighting; cities are embedded in the plane using multidimensional scaling on the original text embeddings and colored according to the legend in Figure 10. Right: as in the left panel, but with only the peel of the embedding displayed, using the same coordinates. The maximum-diversity distribution on the emebdding is indicated by radii of the inner disks.

References

[1] A. Boukerche, L. Zheng, and O. Alfandi. Outlier detection: methods, models, and classification. *ACM Computing Surveys (CSUR)*, 53(3):1–37, 2020.

[2] E. Bunch, D. Dickinson, J. Kline, and G. Fung. Practical applications of metric space magnitude and weighting vectors. *arXiv preprint arXiv:2006.14063*, 2020.

[3] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: a survey. *ACM Computing Surveys (CSUR)*, 41(3):1–58, 2009.

[4] K. Devriendt. *Graph Geometry from Effective Resistances*. PhD thesis, University of Oxford, 2022.

[5] E. Deza and M. M. Deza. *Encyclopedia of Distances*. Springer, 2009.

[6] P. Farrelly. *Dumb and Dumber*. New Line Cinema, 1994.

[7] P. Hjorth, P. Lisoněk, S. Markvorsen, and C. Thomassen. Finite metric spaces of strictly negative type. *Linear Algebra and its Applications*, 270(1-3):255–273, 1998.

[8] S. Huntsman. Diversity enhancement via magnitude. In *International Conference on Evolutionary Multi-Criterion Optimization*, 2023.

[9] S. Huntsman. Magnitude of arithmetic scalar and matrix categories. *arXiv preprint arXiv:2304.08334*, 2023.

[10] S. Huntsman. Quality-diversity in dissimilarity spaces. In *Genetic and Evolutionary Computation Conference*, 2023.

[11] R. Krauthgamer and Y. Rabani. Improved lower bounds for embeddings into l_1. *SIAM Journal on Computing*, 38(6):2487–2498, 2009.

[12] T. Leinster. *Entropy and Diversity*. Cambridge, 2021.

[13] T. Leinster and C. A. Cobbold. Measuring diversity: the importance of species similarity. *Ecology*, 93(3):477–489, 2012.

[14] T. Leinster and M. W. Meckes. Maximizing diversity in biology and beyond. *Entropy*, 18(3):88, 2016.

[15] Y.-F. Lo and S.-W. Hsiao. Exploring the semantic representations of text in subspaces of latent space: A case study on color. In *2024 IEEE International Conference on Big Data (BigData)*, pages 8765–8767. IEEE, 2024.

[16] S. Mahabadi and S. Trajanovski. Core-sets for fair and diverse data summarization. *Advances in Neural Information Processing Systems*, 2023.

[17] M. W. Meckes. Positive definite metric spaces. *Positivity*, 17(3):733–757, 2013.

[18] M. W. Meckes. Magnitude, diversity, capacities, and dimensions of metric spaces. *Potential Analysis*, 42:549–572, 2015.

[19] R. Munroe. Color survey results. `https://blog.xkcd.com/2010/05/03/color-survey-results/`, 2010.

[20] H. Needham. *The Cannonball Run*. 20th Century Fox, 1981.

[21] C. R. Rao. Convexity properties of entropy functions and analysis of diversity. In *Inequalities in Statistics and Probability*, volume 5, pages 68–78. Institute of Mathematical Statistics, 1984.

[22] F. Rubel, K. Brugger, K. Haslinger, and I. Auer. The climate of the European Alps: shift of very high resolution Köppen-Geiger climate zones 1800–2100. *Meteorologische Zeitschrift*, 26(2):115–125, 2017.

[23] B. Warner, A. Chaffin, B. Clavié, O. Weller, O. Hallström, S. Taghadouini, A. Gallagher, R. Biswas, F. Ladhak, T. Aarsen, et al. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. *arXiv preprint arXiv:2412.13663*, 2024.

[24] Wikipedia. List of united states cities by population. `http://en.wikipedia.org/w/index.php?title=List%20of%20United%20States%20cities%20by%20population&oldid=1298495063`, 2025.

[25] S. Willerton. Heuristic and computer calculations for the magnitude of metric spaces. *arXiv preprint arXiv:0910.5500*, 2009.

[26] B. Yates. *Cannonball!: World's Greatest Outlaw Road Race*. Motorbooks International, 2003.

[27] YouGov. How good is "good"? `https://today.yougov.com/society/articles/21717-how-good-good-1`, 2018.

Supplement to: Peeling metric spaces of strict negative type

Steve Huntsman
steve.huntsman@cynnovative.com

A Proofs

A.1 Proof of Theorem 2

Proof. [1] Algorithm 1 terminates as soon as condition i) holds. Each iteration of the while loop requires $O(n^\omega)$ operations and there are at most n iterations, which conditionally establishes the computational complexity bound. Since the condition $e_k^T dp \geq e_j^T dp$ for $j \in \mathcal{J}$ trivially holds at initialization, it therefore suffices to show that this condition is maintained throughout the while loop as $\mathcal{J}$ and p are updated.

For $j \in \mathcal{J}$, we have that (abusively writing e_j for the jth standard basis vector in both $\mathbb{R}^n$ and $\mathbb{R}^{|\mathcal{J}|}$)

$$e_j^T dp = \frac{e_j^T d_{\mathcal{J},\mathcal{J}} d_{\mathcal{J},\mathcal{J}}^{-1} 1_\mathcal{J}}{1_\mathcal{J}^T d_{\mathcal{J},\mathcal{J}}^{-1} 1_\mathcal{J}} = \frac{1}{1_\mathcal{J}^T d_{\mathcal{J},\mathcal{J}}^{-1} 1_\mathcal{J}} =: c.$$

(Note that any nonempty principal submatrix $d_{\mathcal{J},\mathcal{J}}$ of d is strict negative type and hence invertible.) Meanwhile for $k \in \mathcal{J}^c \equiv [n] \backslash \mathcal{J}$,

$$e_k^T dp = \frac{d_{k,\mathcal{J}} d_{\mathcal{J},\mathcal{J}}^{-1} 1_\mathcal{J}}{1_\mathcal{J}^T d_{\mathcal{J},\mathcal{J}}^{-1} 1_\mathcal{J}} = c \cdot \alpha_k,$$

where $\alpha_k := d_{k,\mathcal{J}} d_{\mathcal{J},\mathcal{J}}^{-1} 1_\mathcal{J}$. We need to show that $\alpha_k \leq 1$ for $k \in \mathcal{J}^c$.

Accordingly, write $\mathcal{J}' := \mathcal{J} \cup \{k\}$ and

$$d_{\mathcal{J}',\mathcal{J}'} = \begin{pmatrix} d_{\mathcal{J},\mathcal{J}} & d_{\mathcal{J},k} \\ d_{k,\mathcal{J}} & 0 \end{pmatrix}$$

so that the Schur complement is $S := d_{\mathcal{J}',\mathcal{J}'} / d_{\mathcal{J},\mathcal{J}} = -d_{k,\mathcal{J}} d_{\mathcal{J},\mathcal{J}}^{-1} d_{\mathcal{J},k}$. Meanwhile, the Schur complement formula yields

$$S = \frac{\det d_{\mathcal{J}',\mathcal{J}'}}{\det d_{\mathcal{J},\mathcal{J}}}.$$

Since both $d_{\mathcal{J}',\mathcal{J}'}$ and $d_{\mathcal{J},\mathcal{J}}$ are strict negative type, each has exactly one positive eigenvalue, with the remainder negative. Therefore $S < 0$.

Block inverting, we have that

$$d_{\mathcal{J}',\mathcal{J}'}^{-1} = \begin{pmatrix} d_{\mathcal{J},\mathcal{J}}^{-1} + d_{\mathcal{J},\mathcal{J}}^{-1} d_{\mathcal{J},k} S^{-1} d_{k,\mathcal{J}} d_{\mathcal{J},\mathcal{J}}^{-1} & -d_{\mathcal{J},\mathcal{J}}^{-1} d_{\mathcal{J},k} S^{-1} \\ -S^{-1} d_{k,\mathcal{J}} d_{\mathcal{J},\mathcal{J}}^{-1} & S^{-1} \end{pmatrix}.$$

[1] This is a substantial revision of an earlier proof that had an error in the second claimed equality in a chain of equalities near the end. This revision draws on a session with Claude Opus at `https://claude.ai/share/65e7edc0-c0c3-4c0b-915c-36179b719dfe`.

This yields that

$$e_k^T d_{\mathcal{J}',\mathcal{J}'}^{-1} 1_{\mathcal{J}'} = -S^{-1} d_{k,\mathcal{J}} d_{\mathcal{J},\mathcal{J}}^{-1} 1_{\mathcal{J}} + S^{-1} = S^{-1}(1-\alpha_k).$$

Since $S^{-1} < 0$, we now have $e_k^T d_{\mathcal{J}',\mathcal{J}'}^{-1} 1_{\mathcal{J}'} < 0 \iff \alpha_k < 1$. Suppose instead that $\alpha_k \geq 1$: then $e_k^T d_{\mathcal{J}',\mathcal{J}'}^{-1} 1_{\mathcal{J}'} \geq 0$.

Applying this observation repeatedly for each element removed from a preceding instance of $\mathcal{J}$ in the while loop shows that k would not have been removed from $\mathcal{J}$ in the preceding iteration, contradicting the assumption $k \in \mathcal{J}^c$. □

A.2 Proof of Lemma 1

Proof. We have that

$$\begin{aligned} x^T(d^{(1)} +_1 d^{(2)})x &= x^T(d^{(1)} \otimes J^{(2)} + J^{(1)} \otimes d^{(2)})x \\ &= (x^{(1)})^T d^{(1)} x^{(1)} + (x^{(2)})^T d^{(2)} x^{(2)} \end{aligned} \tag{1}$$

with $x^{(1)} := (I^{(1)} \otimes (1^{(2)})^T)x$ and $x^{(2)} := ((1^{(1)})^T \otimes I^{(2)})x$. (That is, $x_{j_1}^{(1)} = \sum_{j_2} x_{j_1 j_2}$ and $x_{j_2}^{(2)} = \sum_{j_1} x_{j_1 j_2}$.) Now

$$(1^{(1)})^T x^{(1)} = ((1^{(1)})^T \otimes (1^{(2)})^T)x = (1^{(2)})^T x^{(2)},$$

so

$$((1^{(1)})^T \otimes (1^{(2)})^T)x = 0 \iff (1^{(1)})^T x^{(1)} = 0 = (1^{(2)})^T x^{(2)}.$$

It follows from (1) that if $1^T x = 0$, then $x^T(d^{(1)} +_1 d^{(2)})x = 0$. That is, $d^{(1)} +_1 d^{(2)}$ is negative type, but not strict negative type. □

A.3 Proof of Theorem 3

Proof. Let $d^{(1)}$ and $d^{(2)}$ be finite strict negative type, and let $x^T x = 1$ with $1^T x = 0$. Now

$$\begin{aligned} x^T\left(d^{(1)} +_p d^{(2)}\right)x &= \sum_{j_1 j_2 k_1 k_2} x_{j_1 j_2}\left(d^{(1)} +_p d^{(2)}\right)_{(j_1,j_2),(k_1,k_2)} x_{k_1 k_2} \\ &= \sum_{\substack{j_1 j_2 k_1 k_2 \\ j_1 \neq k_1 \\ j_2 \neq k_2}} + \sum_{\substack{j_1 j_2 k_1 k_2 \\ j_1 \neq k_1 \\ j_2 = k_2}} + \sum_{\substack{j_1 j_2 k_1 k_2 \\ j_1 = k_1 \\ j_2 \neq k_2}} - \sum_{\substack{j_1 j_2 k_1 k_2 \\ j_1 = k_1 \\ j_2 = k_2}} \end{aligned}$$

where on the second line we engage in the notational abuse of suppressing summands for the sake of overall clarity.

The last sum above is identically zero, and by assumption the second and third sums are respectively $x^{(1)T} d^{(1)} x^{(1)}$ and $x^{(2)T} d^{(2)} x^{(2)}$, where here we use notation introduced in the proof of the preceding lemma. That proof also shows that these two sums are zero since $1^T x = 0$. It follows that

$$\begin{aligned} x^T\left(d^{(1)} +_p d^{(2)}\right)x &= \sum_{\substack{j_1 j_2 k_1 k_2 \\ j_1 \neq k_1 \\ j_2 \neq k_2}} x_{j_1 j_2}\left(d^{(1)} +_p d^{(2)}\right)_{(j_1,j_2),(k_1,k_2)} x_{k_1 k_2} \\ &< \sum_{\substack{j_1 j_2 k_1 k_2 \\ j_1 \neq k_1 \\ j_2 \neq k_2}} x_{j_1 j_2}\left(d^{(1)} +_1 d^{(2)}\right)_{(j_1,j_2),(k_1,k_2)} x_{k_1 k_2} \\ &= x^T\left(d^{(1)} +_1 d^{(2)}\right)x \\ &= 0, \end{aligned}$$

where the strict inequality on the second line above holds because all of the terms $\left(d^{(1)} +_p d^{(2)}\right)_{(j_1,j_2),(k_1,k_2)}$ and $\left(d^{(1)} +_1 d^{(2)}\right)_{(j_1,j_2),(k_1,k_2)}$ are nonzero, and because $(a_1^p + a_2^p)^{1/p} < a_1 + a_2$ for $a_1, a_2 > 0$ and $p > 1$.

The preceding proposition and lemma complete the proof. □

B The non-strict negative type case

The problem of maximizing the quadratic entropy $p^T dp$ over Δ_{n-1} has been considered in, e.g., [6, 8, 13, 9]. In the Euclidean setting, [13] points out that this maximum quadratic entropy is realized by the squared radius of a minimal sphere containing points with distance matrix $\sqrt{d}$ (which is also a Euclidean distance matrix); the support of p corresponds to the subset of points on this sphere.

It can be shown (see, e.g., Example 5.16 of [4]) that maximizing $p^T d' p$ over Δ_{n-1} is $\mathbf{NP}$-hard for arbitrary d'. This is unsurprising since quadratic programming is $\mathbf{NP}$-hard [17] and remains so even when the underlying matrix has only one eigenvalue with a given sign [11]. While this sign condition is typical for Euclidean distance matrices [18, 1], it nevertheless turns out that such matrices are strict negative type, so as mentioned in the main text, $p^T dp$ is convex and Z is positive semidefinite for *all* $t > 0$.

It appears likely that maximizing $p^T dp$ over Δ_{n-1} is still generally $\mathbf{NP}$-hard when Z is positive definite only for all sufficiently small t. Yet in this intermediate case we can still do slightly better than despairing at general intractability or resorting to Algorithm 1 as a heuristic of uncertain effectiveness by developing a nontrivial bound. By Theorem 3.2 of [9] we have that

$$\arg \max_{p \in \mathbb{R}^n : 1^T p = 1} p^T dp = \frac{d^{-1}1}{1^T d^{-1} 1},$$

though in general this extremum (which also equals the limiting weighting $\lim_{t \downarrow 0} Z^{-1}1$) will have negative components. Thus the most practical recourse when Z is positive definite only for all sufficiently small t is to bound $p^T dp$ using

$$\max_{p \in \Delta_{n-1}} p^T dp \leq \max_{p \in \mathbb{R}^n : 1^T p = 1} p^T dp = \frac{1}{1^T d^{-1} 1}. \tag{2}$$

This unpacks as

$$\begin{aligned} \lim_{t \downarrow 0} \frac{\log D_1^Z(p)}{\max_{p \in \Delta_{n-1}} \log D_1^Z(p)} &\geq \lim_{t \downarrow 0} \frac{\log D_1^Z(p)}{\max_{p \in \mathbb{R}^n : 1^T p = 1} \log D_1^Z(p)} \\ &= (p^T dp) \cdot (1^T d^{-1} 1). \end{aligned} \tag{3}$$

C Visualizing the rainbow

Figure 1 shows the result of considering the entire rainbow at once in the manner of Figures 3-6 of the main text.

D Testing for (strict) negative type

A simple way to check of d is (strict) negative type is to consider the matrix

$$T_k^- := P_k \left(1 e_k^T d + d e_k 1^T - d \right) P_k^T$$

where

$$P_k := \begin{pmatrix} I_k & 0 & 0 \\ 0 & 0 & I_{n-k-1} \end{pmatrix}$$

is the result of deleting the kth row from the identity matrix I_n, and where e_k is the kth column of I_n. Lemma 3.5 of [6] states that d is strict negative type (resp., negative type) iff T_k^- is positive definite (resp., positive semidefinite) for any (and hence all) k. [2]

Another test is presented at the end of §2.2 of [20] (see also [5]): d is strict negative type iff the value of the optimization problem CRQopt(A, b, C) defined therein is positive with $A = -d$, $b = 0$, and $C = 1$. This value can also be computed through standard linear algebra operations.

[2] The matrix $T_{(k)}^-$ has entries of the form $d_{kj} + d_{ik} - d_{ij}$, but without the original kth row and column. See also Lemma 1.7 of [12] as rephrased in Theorem 4.2 of [15], or Theorem 2.1 of [19]. A weaker condition than negative type is *positive (semi)definite*, which amounts to $\exp[-d]$ being a positive (semi)definite matrix [10]. If $\exp[-td]$ is positive (semi)definite for every $t > 0$, then d is called *stably positive (semi)definite*. Theorem 3.3 of [10] shows that stable positive definiteness and negative type are equivalent conditions.

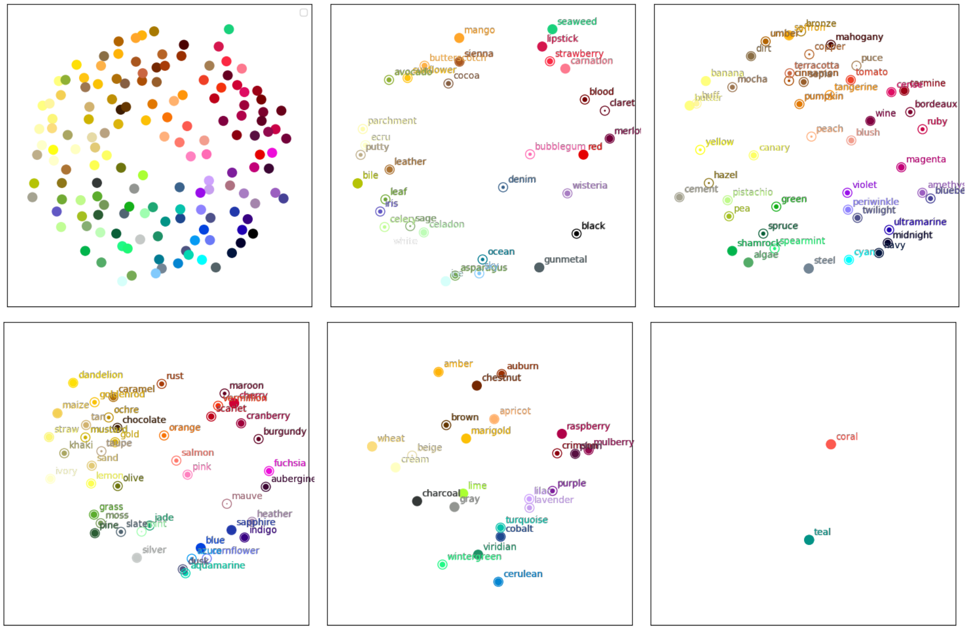

Figure 1: Upper left: response embeddings for 150 colors in the manner of Figures 3 and 5 of the main text. Successive panels: peels of residual “unpeeled” sets. The medoid is in the third peel, and corresponds to a response for “ochre.” Many humans would consider this a very average color, yet the center of the two-dimensional embedding shows more shades close to orange and green, though the overall (persistent) topology of the embedding is arguably annular. Regarding the final peel, it is interesting that teal and coral are complementary colors. This observation motivated the separation of colors in Figures 3-6 of the main text.

In short, checking for (strict) negative type is not a computational bottleneck, though it is expensive enough to be avoided if possible. Theorem 3 gives a result that allows us to avoid a computational check for certain ways of combining strict negative type metrics.

E The most prominent paths depicted in Figures 10 and 11 of the main text

Table 1 shows the most prominent paths depicted in Figures 10 and 11 of the main text.

Table 1: Most heavily weighted paths in the peel shown in Figures 10 and 11 of the main text

Route		
Stop 1	Stop 2	Relative weighting
Lincoln, NE	Aurora, CO	1.0
Tulsa, OK	Oklahoma City, OK	0.63
Aurora, CO	Henderson, NV	0.60
Lincoln, NE	Henderson, NV	0.56
Jersey City, NJ	Newark, NJ	0.55
Aurora, CO	North Las Vegas, NV	0.50
Indianapolis, IN	Tulsa, OK	0.48
Lexington, KY	Louisville, KY	0.45

F An alternative approach to multiobjective path diversity

The method in §4.2 of the main text of selecting paths that cross thresholds for objectives and then selecting a subset with maximal feature diversity is simple but relatively inflexible: for example, the paths must all have the same number of arcs. There is a more scalable and flexible iterative approach that allows for paths with different numbers of arcs, albeit at the cost of added complexity. The basic idea is to keep track of a set of prior diverse paths (initially a single path) and use the enhanced multi-objective A* (EMOA*) algorithm of [16].

In detail, let E be the negative differential magnitude computed relative to prior paths using an edit distance; let $F_1, \dots F_M$ be functions on paths of the form

$$F_m(v_1, \dots, v_\ell) = \sum_{k=1}^{\ell-1} f_m(v_k, v_{k+1}),$$

and let $F'_1, \dots, F'_{M'}$ be of the form

$$F'_{m'}(v_1, \dots, v_\ell) = \max_k f'_m(v_k, v_{k+1}).$$

Abusively writing $\mathrm{Par}(\cdot)$ for a Pareto frontier, we can form $\mathrm{Par}(\mathrm{Par}(F), E \oplus F')$, i.e., the $(E \oplus F')$-Pareto optimal subset of the Pareto frontier of F, where $\oplus$ indicates a concatenation or direct sum. This is relatively easy since EMOA* can approximate $\mathrm{Par}(F)$. Starting with a single F'-optimal path in $\mathrm{Par}(F)$, we can iteratively adjoin distinct paths in $\mathrm{Par}(F)$ that are $(E \oplus F')$-Pareto optimal.

However, E depends on a choice of scale. We can address this while keeping computations tractable by using a cutoff scale [7] that has been computed for prior paths. Still, this scale will increase and the computational cost will as well, so for these reasons among others it may be appropriate to have a finite buffer for prior paths. Moreover, it will often be preferable to enforce a strict negative type metric and work at scale zero. We can always use the ultrametric induced by single linkage clustering [3, 2, 14].

References

[1] E. Bogomolny, O. Bohigas, and C. Schmit. Distance matrices and isometric embeddings. *arXiv preprint arXiv:0710.2063*, 2007.

[2] G. Carlsson and F. Mémoli. Classifying clustering schemes. *Foundations of Computational Mathematics*, 13:221–252, 2013.

[3] G. E. Carlsson and F. Mémoli. Characterization, stability and convergence of hierarchical clustering methods. *Journal of Machine Learning Research*, 11(Apr):1425–1470, 2010.

[4] K. Devriendt. *Graph Geometry from Effective Resistances*. PhD thesis, University of Oxford, 2022.

[5] W. Gander, G. H. Golub, and U. Von Matt. A constrained eigenvalue problem. *Linear Algebra and its Applications*, 114:815–839, 1989.

[6] P. Hjorth, P. Lisoněk, S. Markvorsen, and C. Thomassen. Finite metric spaces of strictly negative type. *Linear Algebra and its Applications*, 270(1-3):255–273, 1998.

[7] S. Huntsman. Diversity enhancement via magnitude. In *International Conference on Evolutionary Multi-Criterion Optimization*, 2023.

[8] J. Izsák and L. Szeidl. Quadratic diversity: its maximization can reduce the richness of species. *Environmental and Ecological Statistics*, 9:423–430, 2002.

[9] S. Izumino and N. Nakamura. Maximization of quadratic forms expressed by distance matrices. *Hokkaido Mathematical Journal*, 35(3):641–658, 2006.

[10] M. W. Meckes. Positive definite metric spaces. *Positivity*, 17(3):733–757, 2013.

[11] P. M. Pardalos and S. A. Vavasis. Quadratic programming with one negative eigenvalue is $\mathbf{NP}$-hard. *Journal of Global Optimization*, 1(1):15–22, 1991.

[12] K. R. Parthasarathy and K. Schmidt. *Positive Definite Kernels, Continuous Tensor Products, and Central Limit Theorems of Probability Theory*. Springer, 1972.

[13] S. Pavoine, S. Ollier, and D. Pontier. Measuring diversity from dissimilarities with Rao's quadratic entropy: are any dissimilarities suitable? *Theoretical Population Biology*, 67(4):231–239, 2005.

[14] R. Rammal, G. Toulouse, and M. A. Virasoro. Ultrametricity for physicists. *Reviews of Modern Physics*, 58(3):765, 1986.

[15] C. R. Rao. Convexity properties of entropy functions and analysis of diversity. In *Inequalities in Statistics and Probability*, volume 5, pages 68–78. Institute of Mathematical Statistics, 1984.

[16] Z. Ren, R. Zhan, S. Rathinam, M. Likhachev, and H. Choset. Enhanced multi-objective A* using balanced binary search trees. In *International Symposium on Combinatorial Search*, 2022.

[17] S. Sahni. Computationally related problems. *SIAM Journal on Computing*, 3(4):262–279, 1974.

[18] I. J. Schoenberg. On certain metric spaces arising from Euclidean spaces by a change of metric and their imbedding in Hilbert space. *Annals of Mathematics*, 38(4):787–793, 1937.

[19] J. H. Wells and L. R. Williams. *Embeddings and Extensions in Analysis*. Springer, 1975.

[20] Y. Zhou, B. Zhaojun, and L. Ren-Cang. Linear constrained Rayleigh quotient optimization: theory and algorithms. *CSIAM Transactions on Applied Mathematics*, 2(2):195–262, 2021.