Soft Matching Distance: A metric on neural representations that captures single-neuron tuning

Meenakshi Khosla
McGovern Institute for Brain Research
Massachusetts Institute for Technology
mkhosla@mit.edu

Alex H. Williams
Center for Neural Science, New York University
Center for Computational Neuroscience
Flatiron Institute
alex.h.williams@nyu.edu

Editors: Marco Fumero, Emanuele Rodolà, Clementine Domine, Francesco Locatello, Gintare Karolina Dziugaite, Mathilde Caron

Abstract

Common measures of neural representational (dis)similarity are designed to be insensitive to rotations and reflections of the neural activation space. Motivated by the premise that the tuning of individual units may be important, there has been recent interest in developing stricter notions of representational (dis)similarity that require neurons to be individually matched across networks. When two networks have the same size (i.e. same number of neurons), a distance metric can be formulated by optimizing over neuron index permutations to maximize tuning curve alignment. However, it is not clear how to generalize this metric to measure distances between networks with different sizes. Here, we leverage a connection to optimal transport theory to derive a natural generalization based on "soft" permutations. The resulting metric is symmetric, satisfies the triangle inequality, and can be interpreted as a Wasserstein distance between two empirical distributions. Further, our proposed metric avoids counter-intuitive outcomes suffered by alternative approaches, and captures complementary geometric insights into neural representations that are entirely missed by rotation-invariant metrics.

1 Introduction

Neural representations of stimuli and actions are often described in terms of "tuning curves" of individual neurons. The most classic example is work by Hubel and Wiesel [1, 2], which found that neurons in the primary visual cortex (V1) of cats were selectively responsive—i.e. "tuned"—to edges with particular orientations. However, the utility of tuning curves is less certain in higher-order brain regions involved in navigation, decision-making, and complex sensory processing. In such areas, neurons often exhibit complex "mixed selectivity" to multiple sensory or task features [3, 4, 5]. Neuroscientists recurrently debate whether tuning curves are meaningful in these situations, and similar debates have recently arisen in the interpretable artificial intelligence community [6, 7, 8].

The tuning of individual neurons is closely connected to studies of neural geometry [9]. In particular, neural tuning curves determine the geometry of population-level neural representations, but the same geometry can be produced by many different sets of tuning curves (Fig. 1A-D). Despite this connection, most investigations implicitly ignore tuning by considering rotation-invariant quantities. Indeed, popular measures of representational (dis)similarity between neural networks—centered kernel alignment (CKA; [10]), canonical correlations analysis (CCA; [11]), representational similarity analysis (RSA; [12]), and Procrustes shape distance [13]—are all invariant to rotations of the neural activation space (such as Fig. 1B vs. 1C). Thus, to study the importance of individual neural tuning (or lack thereof) we require complementary metrics that are sensitive to rotations, but still invariant

Proceedings of the I edition of the Workshop on Unifying Representations in Neural Models (UniReps 2023).

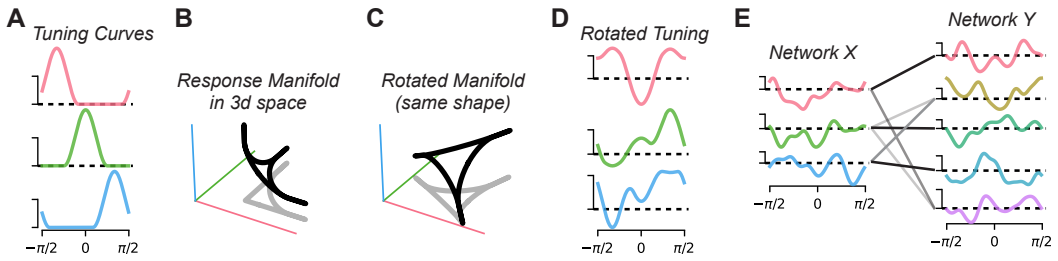

Figure 1: **(A)** Example tuning curves from 3 neurons over a 1D stimulus space. **(B)** Manifold (black curve) arising from tuning curves from panel A in 3D neural firing rate space. Each coordinate axis encodes a single-neural firing rate, colored as in panel A. **(C)** A rotated manifold with the same shape as panel B. **(D)** Tuning curves associated with the rotated manifold in panel C (compare with panel A). **(E)** Schematic illustration of "soft matching." Grayscale lines show matched similar tuning curves across two networks. The darkness of the line indicates the strength of the match.

to permutations of the neuron indices (since such indices are often arbitrary). While a handful of studies have already explored measures that fit these requirements [14, 13], our understanding of these methods and their relation to the more popular approaches cited above is under-developed. For example, Williams et al. [13] proposed a rotation-sensitive, permutation-invariant metric on neural representations based on "permutation Procrustes" analysis [15]. However, their approach suffers a serious limitation—it can only be applied to pairs of networks with the same number of neurons since it relies on a strict one-to-one matching between units. The central contribution of our work is to generalize their metric to the much more typical case of unequal network sizes, which we achieve by leveraging elementary principles from optimal transport theory [16] to obtain a *soft matching* or *soft assignment* [17] between two sets of tuning curves (Fig. 1E). Similar approaches have been used for representation transfer in deep networks [18, 19] and to align word embeddings [20, 21]. Our central motivation—to quantify the reproducibility of tuning curves across networks—is distinct from these prior works, and the details of our approach are suitably adapted where necessary.

Importantly, our approach preserves appealing metric space properties of Williams et al.'s [13] method. Furthermore, we will see that alternative rotation-sensitive measures based on rectangular assignment algorithms [22] and semi-matching [14] can produce unintuitive outcomes that our approach avoids. Finally, we leverage our metric to show that the tuning of individual units is preserved above chance levels in deep layers of artificial and biological networks. Thus, we present a quantitative approach to adjudicate between the competing hypotheses that "tuning matters" vs. "geometry is all you need" [23], and we provide some emprical evidence in favor of the former hypothesis.

2 Methods

We use $\mathcal{O}(N)$ and $\mathcal{P}(N)$ to respectively denote the set of $N \times N$ orthogonal matrices and $N \times N$ permutation matrices. These are commonly referred to as the *orthogonal group* and *permutation group*, respectively. The permutation group is formally defined as follows:

$$\mathcal{P}(N) = \left\{ \boldsymbol{P} \in \mathbb{R}^{N \times N} \;\middle|\; \begin{array}{ll} \sum_i P_{ij} = 1 & \forall\, j \in \{1 \dots N\} \\ \sum_j P_{ij} = 1 & \forall\, i \in \{1 \dots N\} \\ P_{ij} \in \{0, 1\} & \forall\, i, j \in \{1 \dots N\} \times \{1 \dots N\} \end{array} \right\} \tag{1}$$

In words, a "permutation matrix" is a square matrix defined by containing only zeros and ones, and having each row and column sum to one. For any $\boldsymbol{P} \in \mathcal{P}(N)$ and $\boldsymbol{v} \in \mathbb{R}^N$, the matrix-vector multiplication $\boldsymbol{P}\boldsymbol{v}$ outputs a vector with permuted elements. It is easy to check that every permutation matrix is orthogonal, $\boldsymbol{P}^\top \boldsymbol{P} = \boldsymbol{P}\boldsymbol{P}^\top = \boldsymbol{I}$. In other words, $\mathcal{P}(N)$ is a subset of $\mathcal{O}(N)$.

Essential to our approach is the relationship between permutation matrices and the set of *doubly stochastic matrices*. A doubly stochastic matrix is any square, nonnegative matrix whose rows and columns sum to one. The set of doubly stochastic matrices forms a polytope, known as the *Birkoff*

polytope. We denote the N-dimensional Birkhoff polytope as $\mathcal{B}(N)$, formally:

$$\mathcal{B}(N) = \left\{ \boldsymbol{P} \in \mathbb{R}^{N \times N} \;\middle|\; \begin{array}{cc} \sum_i P_{ij} = 1 & \forall\, j \in \{1 \dots N\} \\ \sum_j P_{ij} = 1 & \forall\, i \in \{1 \dots N\} \\ P_{ij} \geq 0 & \forall\, i, j \in \{1 \dots N\} \times \{1 \dots N\} \end{array} \right\} \tag{2}$$

The Birkoff polytope is a convex set. That is, for any two doubly stochastic matrices $\boldsymbol{P}_1 \in \mathcal{B}(N)$ and $\boldsymbol{P}_2 \in \mathcal{B}(N)$, we can define $\boldsymbol{P}_3 = \alpha \boldsymbol{P}_1 + (1 - \alpha)\boldsymbol{P}_2$ for arbitrary $0 \leq \alpha \leq 1$, and be guaranteed that $\boldsymbol{P}_3 \in \mathcal{B}(N)$. The celebrated *Birkhoff–von Neumann theorem* [24, 25] states that the vertices of $\mathcal{B}(N)$ are one-to-one with $\mathcal{P}(N)$, and this relationship will play an important role in our story.

Finally, we will be interested in generalizing the Birkhoff polytope to rectangular matrices, as this will help us generalize permutations to "soft permutations." Specifically, consider a nonnegative matrix $\boldsymbol{P} \in \mathbb{R}^{N_x \times N_y}$ whose rows each sum to $1/N_x$ and whose columns each sum to $1/N_y$. Thus, the sum of all entries is equal to one. The set of all such matrices defines a *transportation polytope* [26]; we denote this set as $\mathcal{T}(N_x, N_y)$ and define it formally:

$$\mathcal{T}(N_x, N_y) = \left\{ \boldsymbol{P} \in \mathbb{R}^{N_x \times N_y} \;\middle|\; \begin{array}{cc} \sum_i P_{ij} = 1/N_y & \forall\, j \in \{1 \dots N_y\} \\ \sum_j P_{ij} = 1/N_x & \forall\, i \in \{1 \dots N_x\} \\ P_{ij} \geq 0 & \forall\, i, j \in \{1 \dots N_x\} \times \{1 \dots N_y\} \end{array} \right\} \tag{3}$$

Note that when $N = N_x = N_y$, all rows and columns sum to $1/N$. Thus, for any $\boldsymbol{P} \in \mathcal{B}(N)$ we have that $(1/N)\boldsymbol{P} \in \mathcal{T}(N, N)$. In other words, except for a minor re-scaling factor, the Birkhoff polytope is a special case of the transportation polytope we defined.

2.1 Problem Setup and Procrustes Distance

To study the problem of comparing neural activations from different networks, we adopt a similar setting described in prior works (e.g. [11, 10, 13]). Specifically, neural population response vectors are sampled from two networks over a set of M stimulus inputs. We collect these responses into two matrices $\boldsymbol{X} \in \mathbb{R}^{M \times N_x}$ and $\boldsymbol{Y} \in \mathbb{R}^{M \times N_y}$, where N_x and N_y denote the respective number of neurons in each network. Our goal is to come up with methods to quantify the (dis)similiarity between $\boldsymbol{X}$ and $\boldsymbol{Y}$ while ignoring nuisance transformations. For example, CKA [10] and Procrustes shape distance [13] are invariant to translations, isotropic scalings, rotations, and reflections as nuisance transformations. Other measures based on linear regression [27], partial least squares [28], and CCA [11, 29] are invariant to a broader class of nuisance transformations: namely, invertible affine transformations within linear subspaces that account for a high fraction of (co)variance.

Consider the Procrustes shape distance as a concrete example. For convenience, we assume that $\boldsymbol{X}$ and $\boldsymbol{Y}$ have been pre-processed so that their columns sum to zero and $\|\boldsymbol{X}\|_F = \|\boldsymbol{Y}\|_F = 1$. Such pre-processing is necessary to remove the effects of translations and isotropic rescalings. Assuming this pre-processing has been imposed and the two networks have the same number of neurons (i.e. $N = N_x = N_y$), then the Procrustes distance, which we denote $d_{\mathcal{O}}$, can be defined as:

$$d_{\mathcal{O}}(\boldsymbol{X}, \boldsymbol{Y}) = \min_{\boldsymbol{Q} \in \mathcal{O}(N)} \|\boldsymbol{X} - \boldsymbol{Y}\boldsymbol{Q}\|_F \tag{4}$$

Intuitively, the Procrustes distance is the minimal Euclidean distance between between $\boldsymbol{X}$ and $\boldsymbol{Y}$ after optimizing over an orthogonal $N \times N$ alignment matrix $\boldsymbol{Q}$. Thus, it is easy to see that the distance is invariant to rotations and reflections of the neural responses in N-dimensional space. Due to this appealing geometric interpretation, eq. (4) is probably the most common definition of Procrustes distance. However, this definition does not apply when $N_x \neq N_y$ since it would require us to subtract two matrices with different numbers of columns. To remedy this, Williams et al. [13] suggested that $\boldsymbol{X}$ and $\boldsymbol{Y}$ could be embedded into the same dimension by principal components analysis or zero-padding. A more elegant solution is to simply define the Procrustes distance in the following manner, which is valid for $N_x \neq N_y$:

$$d_{\mathcal{O}}(\boldsymbol{X}, \boldsymbol{Y}) = \sqrt{\mathrm{Tr}[\boldsymbol{X}^\top \boldsymbol{X}] + \mathrm{Tr}[\boldsymbol{Y}^\top \boldsymbol{Y}] - 2\|\boldsymbol{X}^\top \boldsymbol{Y}\|_*} \tag{5}$$

where $\|\boldsymbol{M}\|_*$ denotes the nuclear norm, or Schatten 1-norm, which is equal to the sum of the singular values of the matrix $\boldsymbol{M}$. Importantly, eqs. (4) and (5) are equivalent when $N_x = N_y$. This fact is well-established, but we provide a self-contained proof in Supplement A.1 for convenience.

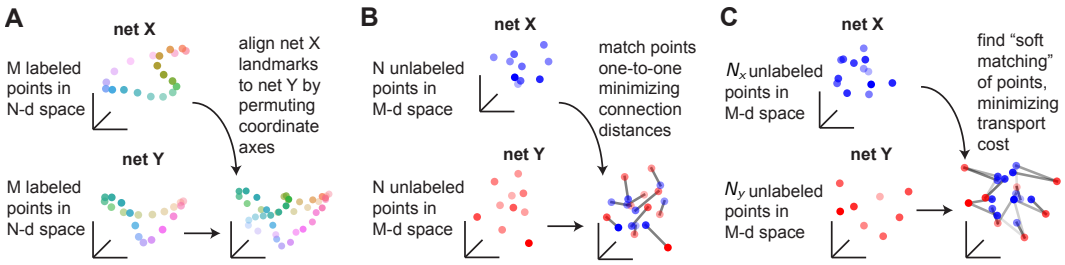

Figure 2: **(A)** One-to-one matching distance, schematized as alignment of M points in N-dimensional space by optimally permuting coordinate axes. Colors denote landmark labels (e.g. image labels) that are common across the two networks. **(B)** Dual perspective of one-to-one matching distance, schematized as matching of N unlabelled points in M-dimensional space. **(C)** Soft matching distance generalizes the picture in panel B, and can be viewed as an optimal transport distance (see main text).

An appealing property of the Procrustes distance is that it defines a *metric space*. More formally, given a class of nuisance transformations $\mathcal{F}$, a distance function d defines a metric space over equivalence classes associated to $\mathcal{F}$ if it satisfies:

$$d(\boldsymbol{X}, \boldsymbol{Y}) = 0 \quad \text{if and only if there exists } f \in \mathcal{F} \text{ such that } \boldsymbol{X} = f(\boldsymbol{Y}) \tag{6}$$

$$d(\boldsymbol{X}, \boldsymbol{Y}) = d(\boldsymbol{Y}, \boldsymbol{X}) \tag{7}$$

$$d(\boldsymbol{X}, \boldsymbol{Y}) \leq d(\boldsymbol{X}, \boldsymbol{Z}) + d(\boldsymbol{Z}, \boldsymbol{Y}) \tag{8}$$

for any $\boldsymbol{X} \in \mathbb{R}^{M \times N_x}$, $\boldsymbol{Y} \in \mathbb{R}^{M \times N_y}$, $\boldsymbol{Z} \in \mathbb{R}^{M \times N_z}$. It can be shown that Procrustes distance satisfies these properties, with $\mathcal{F}$ corresponding to the set of translations, isotropic scalings, rotations, and reflections [13].

2.2 One-to-One Matching Distance

The purpose of this paper is to investigate alternative metrics that are not invariant to general orthogonal transformations (like Procrustes distance), but are still invariant to permutations of the neuron indices. This is motivated by the hypothesis that neurons are usually arbitrarily indexed, but the tuning of individual units may be reproducible across networks.

When comparing networks with the same number of neurons ($N_x = N_y = N$), there is a natural generalization Procrustes distance which we call the *one-to-one matching distance* (Fig. 2A):

$$d_{\mathcal{P}}(\boldsymbol{X}, \boldsymbol{Y}) = \min_{\boldsymbol{P} \in \mathcal{P}(N)} \|\boldsymbol{X} - \boldsymbol{Y}\boldsymbol{P}\|_F \tag{9}$$

The only difference with Procrustes distance is that the minimization is performed over the group of N-dimensional permutation matrices, $\mathcal{P}(N)$, instead of N-dimensional orthogonal matrices, $\mathcal{O}(N)$. Others have referred to this quantity as the "permutation Procrustes" problem [15]. But we prefer one-to-one matching distance to avoid confusion between the two. Like the Procrustes distance, the one-to-one matching distance is symmetric and satisfies the triangle inequality (see, e.g., [13]).

It is well-known, although not immediately obvious, that the optimal permutation can be efficiently found. Indeed, a brute-force enumeration of all $N!$ permutation matrices is impossible for even moderately sized networks, but one can show (see Supplement A.2) that eq. (9) is equivalent to:

$$d_{\mathcal{P}}(\boldsymbol{X}, \boldsymbol{Y}) = \sqrt{\min_{\boldsymbol{P} \in \mathcal{B}(N)} \sum_{i,j} \boldsymbol{P}_{ij} \|\boldsymbol{x}_i - \boldsymbol{y}_j\|_2^2} \tag{10}$$

where the minimization is performed subject to the constraint that $\boldsymbol{P}$ is in the Birkhoff polytope, $\mathcal{B}(N)$, defined in eq. (2). Above, $\|\boldsymbol{x}_i - \boldsymbol{y}_j\|_2^2$ denotes the squared Euclidean distance between column i of $\boldsymbol{X}$ and column j of $\boldsymbol{Y}$. In other words, $\|\boldsymbol{x}_i - \boldsymbol{y}_j\|_2^2$ is a measure of distance between a tuning curve i from network $\boldsymbol{X}$ and tuning curve j from network $\boldsymbol{Y}$.

To build intuition for eq. (10), we offer the following interpretation as a "transportation problem." Specifically, suppose we have N "sender warehouses" at locations $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$, each with one unit of raw material. We would like to transport all of our material to a set of "receiver warehouses"

$\{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N\}$ as cheaply as possible, where $\|\boldsymbol{x}_i - \boldsymbol{y}_j\|_2^2$ quantifies the cost of transporting one unit of material from warehouse i to warehouse j. Any sender warehouse can split its supply amongst multiple receivers, so long as every sender releases all of its supply, $\sum_j \boldsymbol{P}_{ij} = 1$ for all i, and every receiver ends with exactly one unit of supply, $\sum_i \boldsymbol{P}_{ij} = 1$ for all j. The Birkhoff polytope $\mathcal{B}(N)$ represents the set of all valid transportation plans within these constraints. Except in degenerate cases, there is a unique optimum and it is somewhat intuitive that the best strategy forgoes the option to split supplies, and instead finds a one-to-one matching of sender-receiver pairs. That is, the solution will be found at a vertex of the Birkhoff polytope, and therefore be a permutation matrix due to the Birkhoff–von Neumann theorem. Figure 2B provides a schematic illustration of this "dual" view of the one-to-one matching distance (compare with Fig. 2A). Equation (9) can be efficiently computed by linear programming solvers as well as many specialized polynomial-time algorithms [30] (see A.4).

2.3 Soft Matching Distance

The one-to-one matching distance (eqns. 9,10) is not applicable when $N_x \neq N_y$, which crucially limits its utility. In analogy to how eq. (5) adapted the Procrustes distance to handle unequal network sizes, we now seek a similar generalization of the one-to-one matching distance. A natural way to do this is to modify the constraints of the minimization in eq. (10), to obtain:

$$d_{\mathcal{T}}(\boldsymbol{X}, \boldsymbol{Y}) = \sqrt{\min_{\boldsymbol{P} \in \mathcal{T}(N_x, N_y)} \sum_{ij} \boldsymbol{P}_{ij} \|\boldsymbol{x}_i - \boldsymbol{y}_j\|^2} \tag{11}$$

where $\mathcal{T}(N_x, N_y)$ is the transportation polytope defined in eq. (3). The idea is that the transportation and Birkhoff polytopes are essentially equivalent when $N = N_x = N_y$, except for a minor re-scaling factor. In particular, it is easy to verify that $d_{\mathcal{P}}(\boldsymbol{X}, \boldsymbol{Y}) = \sqrt{N} \cdot d_{\mathcal{T}}(\boldsymbol{X}, \boldsymbol{Y})$ when $N_x = N_y = N$. That is, when comparing two networks of equal size, the soft matching distance is equal to the one-to-one matching distance except for a constant factor of $\sqrt{N}$.

Equation (11) involves "soft matching" neuron labels in the sense that every row and every column of the optimal $\boldsymbol{P}$ may have more than one non-zero element. We can again interpret eq. (11) as a transportation problem. In this scenario, we have N_x sender warehouses, each with $1/N_x$ units of material, and N_y receiver warehouses, each requiring $1/N_y$ units. The set of candidate solutions (i.e. feasible transport plans) is given by $\mathcal{T}(N_x, N_y)$. When $N_x \neq N_y$, it is clearly necessary to do some amount of splitting/aggregating of material across multiple receivers/senders to satisfy the constraints of the problem. Figure 2C illustrates this scenario (compare with Fig. 2B).

Readers who are familiar with optimal transport theory [31, 32, 16] will quickly realize that $d_{\mathcal{T}}$ is simply the 2-Wasserstein distance between a uniform mixture of Dirac masses at $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N_x}\}$ and a uniform mixture of Dirac masses at $\{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{N_y}\}$. This allows us to immediately conclude the that soft matching distance is symmetric and satisfies the triangle inequality, which have been cited as advantageous properties [13]. This connection to optimal transport also raises many interesting extensions, such as quantifying representational dissimilarity with entropy-regularized transport divergences [33], but we leave these possibilities to future work.

2.4 Soft Matching Correlation Score and Comparison with Semi-Matching

Suppose that the columns of $\boldsymbol{X}$ and $\boldsymbol{Y}$ have been mean-centered and normalized to unit length. Then $\boldsymbol{x}_i^\top \boldsymbol{y}_j$ is the Pearson correlation between neuron i in network $\boldsymbol{X}$ and neuron j in network $\boldsymbol{Y}$. In this setting, we can formulate a *soft matching correlation score* between two networks:

$$s_{\mathcal{T}}(\boldsymbol{X}, \boldsymbol{Y}) = \max_{\boldsymbol{P} \in \mathcal{T}(N_x, N_y)} \sum_{i,j} \boldsymbol{P}_{ij} \boldsymbol{x}_i^\top \boldsymbol{y}_j \tag{12}$$

When $N_x = N_y$, this has an appealing interpretation: $s_{\mathcal{T}}(\boldsymbol{X}, \boldsymbol{Y})$ equals the average correlation between neurons after optimal one-to-one matching. Clearly, $s_{\mathcal{T}}$ is closely related to the soft matching distance, $d_{\mathcal{T}}$, defined in eq. (11). In fact, one can show (see Supplement A.3) that the matrix $\boldsymbol{P}$ which minimizes the distance in eq. (11) is the same matrix that maximizes the correlation in eq. (12). Although some may prefer to use $d_{\mathcal{T}}$ as a metric satisfying the triangle inequality, others may prefer to interpret $s_{\mathcal{T}}$ since it is normalized between zero and one.

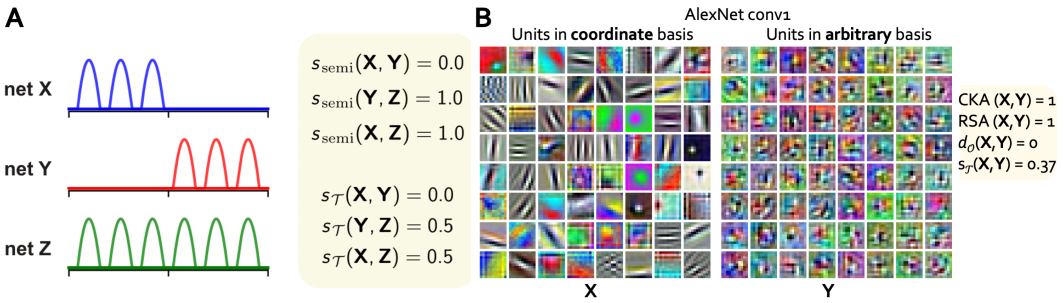

Figure 3: **(A)** (left) Three networks ($\boldsymbol{X}$, $\boldsymbol{Y}$, and $\boldsymbol{Z}$), composed of 1D tuning curves (network sizes: 3, 3, and 6). (right) Similarity scores for networks in panel A quantified by s_{semi} and $s_{\mathcal{T}}$. In this example, each tuning curve is normalized to unit length but not mean-centered. **(B)** Visualization of the 64 `conv1` filters of size $11 \times 11 \times 3$ in AlexNet trained on ImageNet in (left) standard and (right) rotated basis. The rotation matrix is sampled uniformly from the SO(64) group.

How does the soft matching correlation score compare with alternatives? Inspired by analyses in Li et al. [14], we consider a similarity score based on "semi-matching" assignments:

$$s_{\text{semi}}(\boldsymbol{X}, \boldsymbol{Y}) = \frac{1}{N_x} \sum_{i=1}^{N_x} \max_{j \in \{1,\ldots,N_y\}} \boldsymbol{x}_i^\top \boldsymbol{y}_j \tag{13}$$

which is essentially the average correlation after matching every neuron in $\boldsymbol{X}$ to its most similar partner in $\boldsymbol{Y}$. Thus, each neuron in $\boldsymbol{Y}$ may be matched to multiple neurons in $\boldsymbol{X}$ or matched to no partners at all. Alternatively, assuming that $N_y \geq N_x$, one could define:

$$\mathcal{R}(N_x, N_y) = \left\{ \boldsymbol{P} \in \mathbb{R}^{N_x \times N_y} \;\middle|\; \begin{array}{ll} \sum_i P_{ij} \leq 1 & \forall\, j \in \{1 \ldots N_y\} \\ \sum_j P_{ij} = 1 & \forall\, i \in \{1 \ldots N_x\} \\ P_{ij} \in \{0, 1\} & \forall\, i, j \in \{1 \ldots N_x\} \times \{1 \ldots N_y\} \end{array} \right\} \tag{14}$$

as the set of allowable matchings. Here, every neuron in $\boldsymbol{X}$ is matched to one neuron in $\boldsymbol{Y}$, and each neuron in $\boldsymbol{Y}$ is matched to one or zero neurons in $\boldsymbol{X}$. Then, we can define:

$$s_{\mathcal{R}}(\boldsymbol{X}, \boldsymbol{Y}) = \max_{\boldsymbol{P} \in \mathcal{R}(N_x, N_y)} \frac{1}{N_x} \sum_{i,j} \boldsymbol{P}_{ij} \boldsymbol{x}_i^\top \boldsymbol{y}_j \tag{15}$$

as a similarity score. Again, this only applies when $N_y \geq N_x$, since there are no feasible matchings when $N_x > N_y$. Crouse [34] describes algorithms for solving the maximization in eq. (15).

It is easy to see that of the three similarity scores, $s_{\mathcal{T}}$, s_{semi}, and $s_{\mathcal{R}}$, only the soft matching score is symmetric $s_{\mathcal{T}}(\boldsymbol{X}, \boldsymbol{Y}) = s_{\mathcal{T}}(\boldsymbol{Y}, \boldsymbol{X})$. Moreover, both s_{semi} and $s_{\mathcal{R}}$ will tend to view a network with many units (e.g. a "wide" deep net layer) as similar to everything it is compared with. This is illustrated in Figure 3A, which gives a simple example where $\boldsymbol{X}$ and $\boldsymbol{Y}$ are decorrelated, $s_{\text{semi}}(\boldsymbol{X}, \boldsymbol{Y}) = 0$, but at the same time $\boldsymbol{X}$ and $\boldsymbol{Y}$ are both maximally correlated to a third network, $s_{\text{semi}}(\boldsymbol{X}, \boldsymbol{Z}) = s_{\text{semi}}(\boldsymbol{Y}, \boldsymbol{Z}) = 1$. Put differently, on the basis of the s_{semi} similarity scores, it is tempting to observe *"$\boldsymbol{X}$ is perfectly correlated to $\boldsymbol{Z}$, and $\boldsymbol{Y}$ is perfectly correlated to $\boldsymbol{Z}$"* and then falsely conclude that *"$\boldsymbol{X}$ is perfectly correlated to $\boldsymbol{Y}$."* This counter-intuitive behavior also applies to $s_{\mathcal{R}}$, which behaves identically to s_{semi} in this example. In contrast, the soft matching correlation score gives a more intuitive result: it treats $\boldsymbol{Z}$ as only 50% correlated to $\boldsymbol{X}$ and $\boldsymbol{Y}$ (Fig. 3A).

3 Applications

3.1 Highlighting the Limitations of Existing (Dis)similarity Metrics for Single-Neuron Tuning

We first illustrate how prevailing (dis)similarity measures fall short in capturing essential properties of single-neuron tuning due to their rotational invariance. To exemplify this point, we visualize filter weights from the first convolutional layer (`conv1`) of AlexNet trained on ImageNet. Prior research has consistently demonstrated the emergence of Gabor (or "edge-detecting") filters in the initial layers

of deep convolutional networks trained on natural image datasets [35, 36], a phenomenon clearly visible in our analysis (Fig. 3B, left). Intuitively, the structure within these individual filters specifies a coordinate basis in neural activation space that is special and non-random. Indeed, when we apply a random rotation to examine the same weights in an arbitrary basis ($\boldsymbol{Y}$) (Fig. 3B, right), the tuning of individual units differs significantly from that in the coordinate basis ($\boldsymbol{X}$). In particular, we see minimal signatures of edge-detecting filters in $\boldsymbol{Y}$.

It should be emphasized that Fig. 3B visualizes filter weights and applies a random rotation in weight space. Since it is difficult to interpret weights in deeper layers, measures of representational (dis)similarity are usually computed on neural activations instead of weights. However, for the first layer of the network the weights and activations are very closely related: rotating the filter weights (as done in Fig. 3B, *right*) is equivalent to rotating the neural pre-nonlinearity neural activations. Thus, measures like Procrustes distance, CKA, and CCA all fail to capture the distinction between $\boldsymbol{X}$ and $\boldsymbol{Y}$, treating them as equivalent representations. In contrast, the soft-matching distance effectively distinguishes them. Given the sensitivity of the soft-matching distance to single-neuron tuning, beyond mere similarities in representational geometry, we propose that it may also prove valuable in the quantification of disentangled representation learning (see A.5).

3.2 Evidence for privileged coordinate bases in deep hidden layer representations

Figure 3 presents evidence that individual neural tuning functions are non-arbitrary in the first layer of a deep network—a finding that has been documented in past work [35, 36]. The soft-matching distance enables us to precisely quantify this effect and investigate the extent to which the coordinate axes (i.e. individual neural tuning curves) are reproducible in deeper hidden layers. That is, we hypothesize that neural networks trained from different initial weights will converge onto similar tuning curves. This *convergent basis hypothesis* ("tuning matters") can be contrasted with the *arbitrary basis hypothesis* ("geometry is all you need"), which predicts that two networks may converge onto similar representational subspaces but with arbitrarily rotated coordinate axes. As already mentioned, this latter hypothesis is conceptually aligned with existing (dis)similarity measures that are rotation-invariant (see e.g. [10]).

To adjudicate between these hypotheses, we employed the soft-matching similarity metric to explore how the alignment between different neural representations evolves with a gradual change of basis. We compare early and late layer representations in deep convolutional networks trained on object categorization with different random initializations, different architectures (ResNet20/VGG16) [37, 38] and on different datasets (CIFAR10/CIFAR100) [39] to explore whether the bases in different networks share a non-random relationship with respect to each other (evidence for the *convergent basis hypothesis*) or whether this relationship is arbitrary (evidence for the *arbitrary basis hypothesis*). We emphasize that such an inquiry necessitates the development of rotation-sensitive measures, as measures invariant to rotations would exhibit no change with a basis shift.

Our approach is as follows: We sample a random rotation matrix $\boldsymbol{Q}$ uniformly over the special orthogonal group $SO(N)$ [40]. We then instantiate a sequence of rotation matrices that interpolate between $\boldsymbol{Q}$ and the identity matrix ($\boldsymbol{I}$) along the $SO(N)$ manifold. To achieve this, we calculate fractional powers $\boldsymbol{Q}^\alpha = \exp[\alpha \cdot \log[\boldsymbol{Q}]]$, where $0 \leq \alpha \leq 1$ and $\exp[\cdot]$ and $\log[\cdot]$ denote the matrix exponential and matrix logarithm, respectively. Intermediate values of α smoothly interpolate between that $\boldsymbol{Q}$ ($\alpha = 1$) and $\boldsymbol{I}$ ($\alpha = 0$). Thus, varying α smoothly varies the degree of rotation, interpolating between a random basis and the original basis of the neural representation. For each pair of representations $(\boldsymbol{X}, \boldsymbol{Y})$, we alter the basis of one representation, say $\boldsymbol{X}$, by multiplying on the right by $\boldsymbol{Q}^\alpha$. We then measure how the soft-matching correlation score, $s_{\mathcal{T}}(\boldsymbol{X}\boldsymbol{Q}^\alpha, \boldsymbol{Y})$, changes as the degree of rotation is smoothly increased from $\alpha = 0$ to $\alpha = 1$ (Fig. 4A). If the similarity decreases monotonically as a function of α, it furnishes empirical evidence supporting the convergent basis hypothesis.

Our empirical findings are illustrated in Fig. 4B-D. In panel B, we observe that representations from the same network (specifically, ResNet20 trained on CIFAR10) but with varying initial random seeds are significantly more aligned in the standard (coordinate) basis as compared to an arbitrary basis, and this alignment decreases gradually with increasing rotation (α). This trend holds for both early (top) and late (bottom) convolutional layers. This provides evidence that deep convolutional networks trained on image data from different initial weights tend to converge onto similar (though obviously not identical) bases. Strikingly, this trend holds even when comparing networks trained

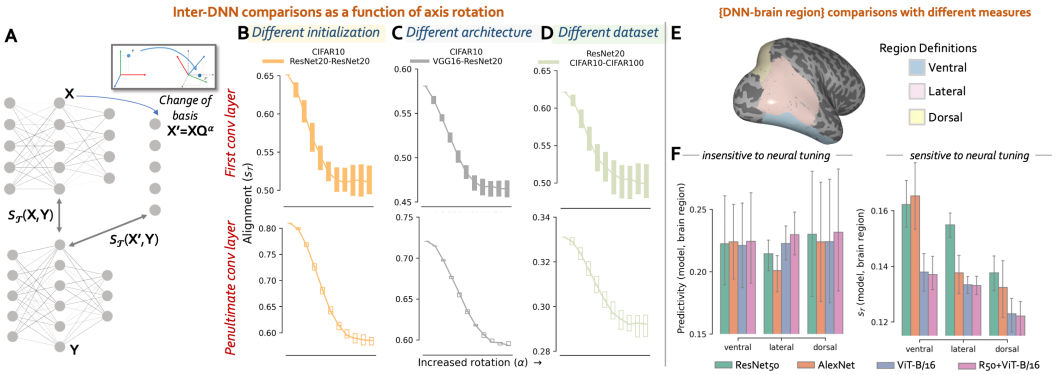

Figure 4: **Soft-matching similarity reveals the non-arbitrariness of activation bases in DNNs.** We investigate how changes in activation bases impact the soft-matching similarity between two neural representations, $\boldsymbol{X}$ and $\boldsymbol{Y}$ (A). Panels B, C and D depict the alignment between networks trained with different initial random seeds (with a fixed ResNet20 architecture), different architectures (ResNet20 vs. VGG16) and different datasets (CIFAR10 vs. CIFAR100) as a function of axis rotation, respectively. **Comparison of DNNs trained on image recognition with the three high-level visual stream representations using different (dis)similarity measures** (E) Definition of the visual streams on a cortical map. (F) DNN-brain region alignment measured using (left) linear predictivity and the soft-matching similarity score ($s_{\mathcal{T}}$).

with different architectures (ResNet20 vs. VGG16, Fig. 4C) as well as networks trained on different datasets (CIFAR10 vs. CIFAR100, Fig. 4D). While the strict, one-to-one matching distance (eq. 9) can be used to quantify these trends across networks with the same architecture, the soft matching distance (eq. 11) or soft matching correlation (eq. 12) developed in this paper are needed to rigorously quantify this effect across different architectures (as in Fig. 4C). Overall, our results suggest that the coordinate axes of neural representations across diverse network architectures and training diets are aligned at above-chance levels, lending support for the *convergent basis hypothesis*.

3.3 Soft-matching distance meaningfully distinguishes between neuroscientific hypotheses

Beyond comparing similarity of representations across different artificial neural networks, there has been a recent surge of interest in applying (dis)similarity measures to compare artificial and biological neural networks [41, 28]. We next demonstrate an application of this metric for model-brain comparisons in neuroscience. In particular, numerous studies over the last decade have revealed that deep networks optimized for behaviorally relevant goals like object categorization learn internal representations that are similar to those in the macaque inferotemporal cortex (IT) (or the ventral visual stream in humans) [27, 42], an area believed to support object recognition in primates [43]. Such findings have sparked optimism that by comparing deep network and neurobiological activation statistics, we can explain specific characteristics of the brain as optimized solutions for specific computational problems faced by organisms [44, 45].

However, recent studies have revealed several counter-intuitive results. For instance, deep networks trained on object categorization were shown to not only accurately model representations in the ventral visual pathway but also in the dorsal and lateral visual pathways [46]. Traditionally, the latter visual streams were hypothesized to be engaged in distinct functions, such as action recognition, social perception, or visually-guided action [47, 48]. Yet another counter-intuitive finding is that vision transformers [49] exhibit equivalent performance to convolutional networks in predicting biological responses [50], even though the latter were designed with some inspiration from biological systems. All of this raises a critical question: are deep networks optimized for object categorization, irrespective of the biological plausibility of their architecture, equally viable models for all three of these visual streams? Or do these findings indicate an inadequacy in our current tools and metrics when it comes to distinguishing between these neuroscientific hypotheses?

One potential explanation is that existing representational (dis)similarity measures may be too permissive to differentiate between various hypotheses. More stringent measures which exclusively seek invariance only with respect to neuron permutations might unveil a different finding. To test this

hypothesis, we leverage the massive Natural Scenes Dataset (NSD) and compare model and fMRI responses across the three visual streams using both a measure that maintains invariance under all invertible linear transformations (linear predictivity, R^2), and our proposed soft-matching distance. We conduct our comparative analyses using the shared set of 1,000 images, each viewed three times by four different participants. We extract feature representations from four candidate neural architectures, all of which were trained for object categorization on ImageNet. This set comprises two DCNNs, namely ResNet50 and AlexNet, as well as two vision transformers [49], ViT-B/16 and a hybrid model (R50+ViT-B/16). In the hybrid model, the input sequence to the ViT is formed from intermediate feature maps of a regular ResNet50. We compare the penultimate representation from each model to the measured brain activity in each of the three high-level visual streams using the soft-matching metrics and linear predictivity. The latter is computed by fitting an l_2 regularized linear regression model on the model representations to predict the measured brain activity using a 70/10/20 train/validation/test split. The predictivity is quantified as the Pearson's correlation coefficient (R) between the measured and predicted responses of each brain voxel, averaged within each stream. The regularization parameter was optimized independently for each subject and each high-level visual stream by testing among 8 log-spaced values in `[1e-4, 1e4]`.

Comparing representations across model and brain region combinations, we observed that linear predictivity (R^2) was insufficient to discern differences between different architectures (CNNs or transformers) and distinct visual processing streams: all model-region scores exhibited similar ranking (Fig. 4F). On the other hand, the soft-matching metrics proved effective in adjudicating between models and revealed significant differences (i) among CNNs vs. transformer architectures in terms of their similarity to brain representations, and (ii) in the ability of object categorization models to capture representations within the three putative visual streams. According to this metric, convolutional networks emerged as superior models for modeling the ventral visual stream when compared to transformers. Furthermore, object categorization models, regardless of their architectural design, demonstrated a better fit with the ventral visual stream as opposed to the other streams. These conclusions are in line with our intuitive understanding of these models and neural systems.

4 Conclusion

We leveraged concepts from optimal transport theory to introduce a metric on neural representations that is rotation-sensitive but permutation-invariant. This metric, which we call *soft matching distance*, generalizes the one-to-one matching distance proposed in [13] to networks of varying sizes by employing *soft permutation* or *soft assignment* of neurons [17, 18, 19, 20, 21].

The soft matching metric reveals structure that is invisible to popular rotation-insensitive measures like CKA [10], RSA [12], and Procrustes distance [13]. For example, our experiments on CIFAR10 and CIFAR100 leverage soft matching distance to show reproducible convergence of activation bases across networks. This convergence holds true across various factors such as initial random seeds, architectural differences, and training diets. These findings are consistent with a variety of anecdotal accounts within the interpretable deep learning literature, such as "curve detector" units [8], shape tuning [7], and object detectors [51]. Despite these examples, most hidden layer units are not (as far as we can tell) semantically meaningful—yet, these non-semantic units are still important for network performance [6]. The soft matching distance therefore addresses an important need to quantitatively compare single unit tuning across networks without reliance on semantic labels.

Moreover, we extend the utility of our metric to the domain of comparing artificial and biological neural networks. We observed that our metric outperforms measures with more inherent invariances, such as linear predictivity, in terms of distinguishing between models. This development offers neuroscientists an additional—and potentially more discerning—tool to interrogate the commonalities and distinctions between biological and artificial networks.

Why might networks have a distinguished and convergent basis, and why is the basis often overlooked in practice? Prior studies justify the use of rotation-invariant metrics since a network layer is arbitrary up to a full-rank matrix multiplication: hence, the subsequent weight matrix can absorb the matrix inverse, rendering the choice of basis immaterial [10]. However, this disregards the fact that nonlinearities, such as Rectified Linear Units (ReLUs), are applied along particular dimensions (i.e. the coordinate axes) within the space of neural activations. These non-linearities might thus act as symmetry-breaking mechanisms that favor certain activation bases over others.

Overall, our work develops a metric, the *soft matching distance*, which complements existing measures of representational similarity like CKA, RSA, and Procrustes shape distance. Relative to these existing rotation-invariant measures, this new distance is better suited to interrogate representations at the level of single-neuron tuning. Our applications of the metric thus far support the view that single neuron tuning is preserved above chance levels across networks, which may be an important clue into the complex computations performed within artificial and biological neural circuits.

Acknowledgements

We thank Nancy Kanwisher and David Alvarez-Melis for insightful discussions and comments on the manuscript.

References

[1] David H Hubel and Torsten N Wiesel. "Receptive fields of single neurones in the cat's striate cortex". *The Journal of physiology* 148.3 (1959), p. 574.

[2] David H Hubel and Torsten N Wiesel. "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex". *The Journal of physiology* 160.1 (1962), p. 106.

[3] Chou P. Hung, Gabriel Kreiman, Tomaso Poggio, and James J. DiCarlo. "Fast Readout of Object Identity from Macaque Inferior Temporal Cortex". *Science* 310.5749 (2005), pp. 863–866.

[4] Kiah Hardcastle, Niru Maheswaranathan, Surya Ganguli, and Lisa M Giocomo. "A multiplexed, heterogeneous, and adaptive code for navigation in medial entorhinal cortex". *Neuron* 94.2 (2017), pp. 375–387.

[5] Stefano Fusi, Earl K Miller, and Mattia Rigotti. "Why neurons mix: high dimensionality for higher cognition". *Current opinion in neurobiology* 37 (2016), pp. 66–74.

[6] Ari S. Morcos, David G.T. Barrett, Neil C. Rabinowitz, and Matthew Botvinick. "On the importance of single directions for generalization". *International Conference on Learning Representations*. 2018.

[7] Dean A Pospisil, Anitha Pasupathy, and Wyeth Bair. "Artiphysiology reveals V4-like shape tuning in a deep network trained for image classification". *eLife* 7 (2018). Ed. by Eilon Vaadia and Joshua I Gold, e38242.

[8] Nick Cammarata, Gabriel Goh, Shan Carter, Ludwig Schubert, Michael Petrov, and Chris Olah. "Curve Detectors". *Distill* (2020). https://distill.pub/2020/circuits/curve-detectors.

[9] Nikolaus Kriegeskorte and Xue-Xin Wei. "Neural tuning and representational geometry". *Nat. Rev. Neurosci.* 22.11 (2021), pp. 703–718.

[10] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. "Similarity of Neural Network Representations Revisited". *Proceedings of the 36th International Conference on Machine Learning*. Ed. by Kamalika Chaudhuri and Ruslan Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. PMLR, 2019, pp. 3519–3529.

[11] Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. "Svcca: Singular vector canonical correlation analysis for deep learning dynamics and interpretability". *Advances in neural information processing systems* 30 (2017).

[12] Nikolaus Kriegeskorte, Marieke Mur, and Peter Bandettini. "Representational similarity analysis - connecting the branches of systems neuroscience". *Front. Syst. Neurosci.* 2 (2008), p. 4.

[13] Alex H. Williams, Erin Kunz, Simon Kornblith, and Scott W. Linderman. "Generalized Shape Metrics on Neural Representations". *Advances in Neural Information Processing Systems*. Vol. 34. 2021.

[14] Yixuan Li, Jason Yosinski, Jeff Clune, Hod Lipson, and John Hopcroft. "Convergent Learning: Do different neural networks learn the same representations?" *Proceedings of the 1st International Workshop on Feature Extraction: Modern Questions and Challenges at NIPS 2015*. Ed. by Dmitry Storcheus, Afshin Rostamizadeh, and Sanjiv Kumar. Vol. 44. Proceedings of Machine Learning Research. Montreal, Canada: PMLR, 2015, pp. 196–212.

[15] John C Gower and Garmt B Dijksterhuis. *Procrustes problems*. Vol. 30. OUP Oxford, 2004.

[16] Gabriel Peyré and Marco Cuturi. "Computational Optimal Transport: With Applications to Data Science". *Foundations and Trends® in Machine Learning* 11.5-6 (2019), pp. 355–607.

[17] Anand Rangarajan, Haili Chui, and Fred L Bookstein. "The softassign procrustes matching algorithm". *Information Processing in Medical Imaging: 15th International Conference, IPMI'97 Poultney, Vermont, USA, June 9–13, 1997 Proceedings 15*. Springer. 1997, pp. 29–42.

[18] Sidak Pal Singh and Martin Jaggi. "Model fusion via optimal transport". *Advances in Neural Information Processing Systems* 33 (2020), pp. 22045–22055.

[19] Xuhong Li, Yves Grandvalet, Rémi Flamary, Nicolas Courty, and Dejing Dou. "Representation transfer by optimal transport". *arXiv preprint arXiv:2007.06737* (2020).

[20] Edouard Grave, Armand Joulin, and Quentin Berthet. "Unsupervised alignment of embeddings with wasserstein procrustes". *The 22nd International Conference on Artificial Intelligence and Statistics*. PMLR. 2019, pp. 1880–1890.

[21] David Alvarez-Melis, Stefanie Jegelka, and Tommi S. Jaakkola. "Towards Optimal Transport with Global Invariances". *Proceedings of Machine Learning Research*. Ed. by Kamalika Chaudhuri and Masashi Sugiyama. Vol. 89. Proceedings of Machine Learning Research. PMLR, 2019, pp. 1870–1879.

[22] David F Crouse. "On implementing 2D rectangular assignment algorithms". *IEEE Trans. Aerosp. Electron. Syst.* 52.4 (2016), pp. 1679–1696.

[23] Geometry is all you need? The importance of representational geometry across brain areas and cognitive processes. Cosyne Workshops 2022. Lisbon, Portugal.

[24] Garrett Birkhoff. "Three observations on linear algebra". *Univ. Nac. Tacuman, Rev. Ser. A* 5 (1946), pp. 147–151.

[25] John von Neumann. "A Certain Zero-sum Two-person Game Equivalent to the Optimal Assignment Problem". *Contributions to the Theory of Games (AM-28), Volume II*. Ed. by Harold William Kuhn and Albert William Tucker. Princeton: Princeton University Press, 1953, pp. 5–12.

[26] Jesús A De Loera and Edward D Kim. "Combinatorics and geometry of transportation polytopes: An update." *Discrete geometry and algebraic combinatorics* 625 (2013), pp. 37–76.

[27] Daniel L. K. Yamins, Ha Hong, Charles F. Cadieu, Ethan A. Solomon, Darren Seibert, and James J. DiCarlo. "Performance-optimized hierarchical models predict neural responses in higher visual cortex". *Proceedings of the National Academy of Sciences* 111.23 (2014), pp. 8619–8624.

[28] Martin Schrimpf, Jonas Kubilius, Michael J Lee, N Apurva Ratan Murty, Robert Ajemian, and James J DiCarlo. "Integrative Benchmarking to Advance Neurally Mechanistic Models of Human Intelligence". *Neuron* (2020).

[29] Mostafa Safaie, Joanna C. Chang, Junchol Park, Lee E. Miller, Joshua T. Dudman, Matthew G. Perich, and Juan A. Gallego. "Preserved neural population dynamics across animals performing similar behaviour". *bioRxiv* (2022).

[30] Rainer Burkard, Mauro Dell'Amico, and Silvano Martello. *Assignment Problems*. Society for Industrial and Applied Mathematics, 2012.

[31] Cédric Villani et al. *Optimal transport: old and new*. Vol. 338. Springer, 2009.

[32] Filippo Santambrogio. "Optimal transport for applied mathematicians". *Birkäuser, NY* 55.58-63 (2015), p. 94.

[33] Jean Feydy, Thibault Séjourné, François-Xavier Vialard, Shun-ichi Amari, Alain Trouve, and Gabriel Peyré. "Interpolating between Optimal Transport and MMD using Sinkhorn Divergences". *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*. Ed. by Kamalika Chaudhuri and Masashi Sugiyama. Vol. 89. Proceedings of Machine Learning Research. PMLR, 2019, pp. 2681–2690.

[34] David F. Crouse. "On implementing 2D rectangular assignment algorithms". *IEEE Transactions on Aerospace and Electronic Systems* 52.4 (2016), pp. 1679–1696.

[35] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "Imagenet classification with deep convolutional neural networks". *Advances in neural information processing systems* 25 (2012).

[36] Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. "An overview of early vision in inceptionv1". *Distill* 5.4 (2020), e00024–002.

[37] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition". *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[38] Karen Simonyan and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition". *arXiv preprint arXiv:1409.1556* (2014).

[39] Alex Krizhevsky, Geoffrey Hinton, et al. "Learning multiple layers of features from tiny images" (2009).

[40] Maris A. Ozols. "How to generate a random unitary matrix". 2009.

[41] David GT Barrett, Ari S Morcos, and Jakob H Macke. "Analyzing biological and artificial neural networks: challenges with opportunities for synergy?" *Current opinion in neurobiology* 55 (2019), pp. 55–64.

[42] Seyed-Mahdi Khaligh-Razavi and Nikolaus Kriegeskorte. "Deep supervised, but not unsupervised, models may explain IT cortical representation". *PLoS computational biology* 10.11 (2014), e1003915.

[43] James J DiCarlo, Davide Zoccolan, and Nicole C Rust. "How does the brain solve visual object recognition?" *Neuron* 73.3 (2012), pp. 415–434.

[44] Daniel LK Yamins and James J DiCarlo. "Using goal-driven deep learning models to understand sensory cortex". *Nature neuroscience* 19.3 (2016), pp. 356–365.

[45] Nancy Kanwisher, Meenakshi Khosla, and Katharina Dobs. "Using artificial neural networks to ask 'why'questions of minds and brains". *Trends in Neurosciences* 46.3 (2023), pp. 240–254.

[46] Dawn Finzi, Daniel LK Yamins, Kendrick Kay, and Kalanit Grill-Spector. "Do deep convolutional neural networks accurately model representations beyond the ventral stream". *2022 Conference on Cognitive Computational Neuroscience*. 2022.

[47] David Pitcher and Leslie G Ungerleider. "Evidence for a third visual pathway specialized for social perception". *Trends in Cognitive Sciences* 25.2 (2021), pp. 100–110.

[48] Melvyn A Goodale and A David Milner. "Separate visual pathways for perception and action". *Trends in neurosciences* 15.1 (1992), pp. 20–25.

[49] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. "An image is worth 16x16 words: Transformers for image recognition at scale". *arXiv preprint arXiv:2010.11929* (2020).

[50] Colin Conwell, Jacob S Prince, Kendrick N Kay, George A Alvarez, and Talia Konkle. "What can 1.8 billion regressions tell us about the pressures shaping high-level visual representation in brains and machines?" *BioRxiv* (2022), pp. 2022–03.

[51] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. "Object detectors emerge in deep scene cnns". *arXiv preprint arXiv:1412.6856* (2014).

[52] Dimitris Bertsimas and John N Tsitsiklis. *Introduction to linear optimization*. Vol. 6. Athena scientific Belmont, MA, 1997.

[53] Yoshua Bengio, Aaron Courville, and Pascal Vincent. "Representation learning: A review and new perspectives". *IEEE transactions on pattern analysis and machine intelligence* 35.8 (2013), pp. 1798–1828.

[54] Cian Eastwood and Christopher KI Williams. "A framework for the quantitative evaluation of disentangled representations". *International conference on learning representations*. 2018.

A Supplementary Results and Comments

A.1 Proof that eqs. (4) and (5) are equivalent when $N_x = N_y = N$

This is the result of a straightforward calculation, exploiting several elementary facts from linear algebra. First, for any matrix $\boldsymbol{A}$ we have that $\|\boldsymbol{A}\|_F^2 = \mathrm{Tr}[\boldsymbol{A}^\top \boldsymbol{A}]$. Second, for any matrices $\{\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}\}$ with appropriate dimensions such that the product $\boldsymbol{ABC}$ is defined, we have that $\mathrm{Tr}[\boldsymbol{ABC}] = \mathrm{Tr}[\boldsymbol{CAB}] = \mathrm{Tr}[\boldsymbol{BCA}]$, which is called the *cyclic trace property*. Finally, for any orthogonal matrix $\boldsymbol{Q} \in \mathcal{O}(N)$ we have $\boldsymbol{Q}^\top \boldsymbol{Q} = \boldsymbol{Q}\boldsymbol{Q}^\top = \boldsymbol{I}$.

With these ingredients we can manipulate the squared Procrustes distance as follows:

$$
\begin{aligned}
d_{\mathcal{O}}^2(\boldsymbol{X}, \boldsymbol{Y}) &= \min_{\boldsymbol{Q} \in \mathcal{O}(N)} \|\boldsymbol{X} - \boldsymbol{Y}\boldsymbol{Q}\|_F^2 \\
&= \min_{\boldsymbol{Q} \in \mathcal{O}(N)} \mathrm{Tr}[\boldsymbol{X}^\top \boldsymbol{X} + \boldsymbol{Q}^\top \boldsymbol{Y}^\top \boldsymbol{Y}\boldsymbol{Q} - 2\boldsymbol{X}^\top \boldsymbol{Y}\boldsymbol{Q}] && (\|\boldsymbol{A}\|_F^2 = \mathrm{Tr}[\boldsymbol{A}^\top \boldsymbol{A}]) \\
&= \min_{\boldsymbol{Q} \in \mathcal{O}(N)} \mathrm{Tr}[\boldsymbol{X}^\top \boldsymbol{X}] + \mathrm{Tr}[\boldsymbol{Q}^\top \boldsymbol{Y}^\top \boldsymbol{Y}\boldsymbol{Q}] - 2\,\mathrm{Tr}[\boldsymbol{X}^\top \boldsymbol{Y}\boldsymbol{Q}] && (\mathrm{Tr}[\cdot] \text{ is linear}) \\
&= \min_{\boldsymbol{Q} \in \mathcal{O}(N)} \mathrm{Tr}[\boldsymbol{X}^\top \boldsymbol{X}] + \mathrm{Tr}[\boldsymbol{Y}^\top \boldsymbol{Y}\boldsymbol{Q}\boldsymbol{Q}^\top] - 2\,\mathrm{Tr}[\boldsymbol{X}^\top \boldsymbol{Y}\boldsymbol{Q}] && (\text{cyclic trace property}) \\
&= \min_{\boldsymbol{Q} \in \mathcal{O}(N)} \mathrm{Tr}[\boldsymbol{X}^\top \boldsymbol{X}] + \mathrm{Tr}[\boldsymbol{Y}^\top \boldsymbol{Y}] - 2\,\mathrm{Tr}[\boldsymbol{X}^\top \boldsymbol{Y}\boldsymbol{Q}] && (\text{orthogonality}) \\
&= \mathrm{Tr}[\boldsymbol{X}^\top \boldsymbol{X}] + \mathrm{Tr}[\boldsymbol{Y}^\top \boldsymbol{Y}] - 2 \max_{\boldsymbol{Q} \in \mathcal{O}(N)} \mathrm{Tr}[\boldsymbol{X}^\top \boldsymbol{Y}\boldsymbol{Q}] && (\text{first two terms are constant}) \\
&= \mathrm{Tr}[\boldsymbol{X}^\top \boldsymbol{X}] + \mathrm{Tr}[\boldsymbol{Y}^\top \boldsymbol{Y}] - 2\|\boldsymbol{X}^\top \boldsymbol{Y}\|_* && (\text{discussed below})
\end{aligned}
$$

as claimed in the main text. The final step is the comes from the celebrated closed form solution to the orthogonal Procrustes problem, which is comprehensively reviewed by Gower and Dijksterhuis [15]. Briefly, the result can be understood by considering the singular value decomposition $\boldsymbol{X}^\top \boldsymbol{Y} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\top$. Then, due to the cyclic trace property,

$$
\mathrm{Tr}[\boldsymbol{X}^\top \boldsymbol{Y}\boldsymbol{Q}] = \mathrm{Tr}[\boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\top \boldsymbol{Q}] = \mathrm{Tr}[\boldsymbol{S}\boldsymbol{V}^\top \boldsymbol{Q}\boldsymbol{U}] \tag{16}
$$

This final expression is maximized by setting $\boldsymbol{Q} = \boldsymbol{V}\boldsymbol{U}^\top$, resulting in:

$$
\mathrm{Tr}[\boldsymbol{S}\boldsymbol{V}^\top \boldsymbol{V}\boldsymbol{U}^\top \boldsymbol{U}] = \mathrm{Tr}[\boldsymbol{S}] = \|\boldsymbol{X}^\top \boldsymbol{Y}\|_* \tag{17}
$$

Since the sum of the diagonal elements of $\boldsymbol{S}$ is simply the sum of the singular values of $\boldsymbol{X}^\top \boldsymbol{Y}$ (i.e. equal to the nuclear norm of this matrix).

A.2 Proof that eqs. (9) and (10) are equivalent

Recall that we are in the setting where $N_x = N_y = N$. First, for any matrix $\boldsymbol{A} \in \mathbb{R}^{M \times N}$ with columns $\{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N\}$ we have $\|\boldsymbol{A}\|_F^2 = \sum_{i=1}^N \|\boldsymbol{a}_i\|^2$. Since $\sum_j \boldsymbol{P}_{ij}\boldsymbol{y}_j$ gives column i of the matrix product $\boldsymbol{Y}\boldsymbol{P}$, we have:

$$
\|\boldsymbol{X} - \boldsymbol{Y}\boldsymbol{P}\|_F^2 = \sum_{i=1}^N \|\boldsymbol{x}_i - \sum_{j=1}^N \boldsymbol{P}_{ij}\boldsymbol{y}_j\|^2 \tag{18}
$$

Recall that $\boldsymbol{P}$ is a permutation matrix in the present context. Thus, let $\sigma(i) \in \{1, \ldots, N\}$ denote the index of the unique nonzero element of row i in $\boldsymbol{P}$. Intuitively, $\sigma(i)$ defines the permutation in which column i of $\boldsymbol{X}$ is matched to column $\sigma(i)$ of $\boldsymbol{Y}$. With this notation, we can re-write the expression above:

$$
\sum_{i=1}^N \|\boldsymbol{x}_i - \sum_{j=1}^N \boldsymbol{P}_{ij}\boldsymbol{y}_j\|^2 = \sum_{i=1}^N \|\boldsymbol{x}_i - \boldsymbol{y}_{\sigma(i)}\|^2 \tag{19}
$$

Now define $\delta[i, j]$ as a function that takes in two integers and evaluates to one if $i = j$ and evaluates to zero if $i \neq j$. (This is often called the Kronecker delta function.) We can write:

$$
\sum_{i=1}^N \|\boldsymbol{x}_i - \boldsymbol{y}_{\sigma(i)}\|^2 = \sum_{i=1}^N \sum_{j=1}^N \delta[\sigma(i), j] \cdot \|\boldsymbol{x}_i - \boldsymbol{y}_j\|^2 \tag{20}
$$

since the inner sum will evaluate to zero whenever $j \neq \sigma(i)$. Finally, we argue that $\boldsymbol{P}_{ij} = \delta[\sigma(i), j]$. Indeed, $\sum_j \delta[\sigma(i), j]\boldsymbol{y}_j = \boldsymbol{y}_\sigma(i)$ which agrees with $\sum_j \boldsymbol{P}_{ij}\boldsymbol{y}_j = \boldsymbol{y}_\sigma(i)$. Thus, we have shown that:

$$d^2_{\mathcal{P}}(\boldsymbol{X}, \boldsymbol{Y}) = \min_{\boldsymbol{P} \in \mathcal{P}(N)} \|\boldsymbol{X} - \boldsymbol{Y}\boldsymbol{P}\|_F^2 = \min_{\boldsymbol{P} \in \mathcal{P}(N)} \sum_{i,j} \boldsymbol{P}_{ij} \|\boldsymbol{x}_i - \boldsymbol{y}_j\|_2^2 \tag{21}$$

To arrive at eq. (10), we need to show that we can relax the constraint of the minimization over the permutation group to over the Birkhoff polytope—i.e. to show that:

$$\min_{\boldsymbol{P} \in \mathcal{P}(N)} \sum_{i,j} \boldsymbol{P}_{ij} \|\boldsymbol{x}_i - \boldsymbol{y}_j\|_2^2 = \min_{\boldsymbol{P} \in \mathcal{B}(N)} \sum_{i,j} \boldsymbol{P}_{ij} \|\boldsymbol{x}_i - \boldsymbol{y}_j\|_2^2 \tag{22}$$

Here we evoke two well-known results. First, the celebrated Birkhoff–von Neumann theorem states that the vertices of $\mathcal{B}(N)$ are one-to-one matched with the permutation matrices $\mathcal{P}(N)$. Second, the final expression is a linear program since the objective function is linear in $\boldsymbol{P}$ and the constraints are linear (as can be verified from eq. 2). Thus, we evoke a basic fact from the theory of linear programming (see e.g. [52]), which states that, assuming that a finite solution exists, at least one vertex of the feasible set is a solution. Any such vertex is called a "basic feasible solution" and the fact that such solutions exist motivates the well known simplex algorithm for linear programming. Thus, we conclude that relaxing the constraints from $\boldsymbol{P} \in \mathcal{P}(N)$ to $\boldsymbol{P} \in \mathcal{B}(N)$ does not allow us to further minimize the objective, and so eq. (22) is valid. Taking square roots on both sides of eq. (22) proves the result claimed in the main text.

A.3 Relation between soft matching distance and correlation score

Here we show that the optimal soft permutation matrix $\boldsymbol{P} \in \mathcal{T}(N_x, N_y)$ that minimizes the expression in eq. 11 equals the one which maximizes the expression in eq. 12. First, beginning with the minimization problem in eq. 11, we can break the expression into three terms:

$$\begin{aligned}
&\operatorname*{argmin}_{\boldsymbol{P} \in \mathcal{T}(N_x, N_y)} \sum_{ij} \boldsymbol{P}_{ij} \|\boldsymbol{x}_i - \boldsymbol{y}_j\|^2 && (23)\\
&= \operatorname*{argmin}_{\boldsymbol{P} \in \mathcal{T}(N_x, N_y)} \sum_{ij} \boldsymbol{P}_{ij} (\boldsymbol{x}_i^\top \boldsymbol{x}_i + \boldsymbol{y}_j^\top \boldsymbol{y}_j - 2\boldsymbol{x}_i^\top \boldsymbol{y}_j) && (24)\\
&= \operatorname*{argmin}_{\boldsymbol{P} \in \mathcal{T}(N_x, N_y)} \underbrace{\sum_{ij} \boldsymbol{P}_{ij} \boldsymbol{x}_i^\top \boldsymbol{x}_i}_{\text{(A)}} + \underbrace{\sum_{ij} \boldsymbol{P}_{ij} \boldsymbol{y}_j^\top \boldsymbol{y}_j}_{\text{(B)}} - 2 \underbrace{\sum_{ij} \boldsymbol{P}_{ij} \boldsymbol{x}_i^\top \boldsymbol{y}_j}_{\text{(C)}} && (25)
\end{aligned}$$

Considering term (A) first, we argue that this term is constant with respect to any feasible $\boldsymbol{P}$ since:

$$\sum_{ij} \boldsymbol{P}_{ij} \boldsymbol{x}_i^\top \boldsymbol{x}_i = \sum_{i=1}^{N_x} \left(\sum_{j=1}^{N_y} \boldsymbol{P}_{ij} \right) \boldsymbol{x}_i^\top \boldsymbol{x}_i = \sum_{i=1}^{N_x} \left(\frac{1}{N_x} \right) \boldsymbol{x}_i^\top \boldsymbol{x}_i \tag{26}$$

where the final equality follows from definition of the transportation polytope in eq. (3)—namely, the rows of $\boldsymbol{P}$ each sum to $1/N_x$. We then can make a similar argument for term (B). In particular, since the columns of $\boldsymbol{P}$ each sum to $1/N_y$, we have:

$$\sum_{ij} \boldsymbol{P}_{ij} \boldsymbol{y}_j^\top \boldsymbol{y}_j = \sum_{j=1}^{N_y} \left(\sum_{i=1}^{N_x} \boldsymbol{P}_{ij} \right) \boldsymbol{y}_j^\top \boldsymbol{y}_j = \sum_{j=1}^{N_y} \left(\frac{1}{N_y} \right) \boldsymbol{y}_j^\top \boldsymbol{y}_j \tag{27}$$

In summary, we see that only term (C) is non-constant. That is, we have

$$\begin{aligned}
\operatorname*{argmin}_{\boldsymbol{P} \in \mathcal{T}(N_x, N_y)} \sum_{ij} \boldsymbol{P}_{ij} \|\boldsymbol{x}_i - \boldsymbol{y}_j\|^2 &= \operatorname*{argmin}_{\boldsymbol{P} \in \mathcal{T}(N_x, N_y)} -2 \sum_{ij} \boldsymbol{P}_{ij} \boldsymbol{x}_i^\top \boldsymbol{y}_j + const. && (28)\\
&= \operatorname*{argmax}_{\boldsymbol{P} \in \mathcal{T}(N_x, N_y)} \sum_{ij} \boldsymbol{P}_{ij} \boldsymbol{x}_i^\top \boldsymbol{y}_j && (29)
\end{aligned}$$

as we have claimed.

A.4 Computational complexity

Computing the soft-matching distance requires solving an optimal transport problem in the discrete setting. The solution to the transportation problem, which is a linear program, can be derived using the network simplex algorithm. With efficient implementations of the simplex algorithm as in the Python Optimal Transport Library, the complexity of solving the linear program is $\mathrm{O}(n^3 \log n)$, assuming that the two representations being compared have n units. Such efficient implementations enable broad application of optimal transport-based solutions in real-world settings.

A.5 Relevance of the Soft-Matching Metric to Disentangled Representation Learning Metrics

It is worth noting that the proposed soft-matching metric can also serve as a valuable tool in the field of disentangled representation learning (DRL) [53] due to its sensitivity to the representational basis. In DRL, the objective is to learn a model that effectively disentangles and makes the underlying generative factors of the data explicit in representational form (i.e. aligned with representational units). Within the DRL literature, various measures have been developed to quantify the alignment between learned representations and ground truth generative factors. Typically, the desiderata involve a combination of different criteria, encompassing the similarity in information content (explicitness) and the degree of one-to-one correspondence between the representational units and generative factors (modularity and compactness) [54]. The soft-matching distance metric offers a unique advantage by simultaneously capturing sensitivity to both of these critical properties.