

Generative Similarity, Generalization, and Abstraction

Raja Marjieh^{1,†}, Sreejan Kumar^{2,3,4,†}, Declan Campbell²,

Liyi Zhang⁵, Jake Snell⁵, and Thomas L. Griffiths^{1,5}

¹Department of Psychology, Princeton University

²Princeton Neuroscience Institute, Princeton University

³Department of Psychology, New York University

⁴Zuckerman Institute, Columbia University

⁵Department of Computer Science, Princeton University

[†]Equal Contribution.

Author Note

Corresponding authors: Raja Marjieh (raja.marjieh@princeton.edu) and
Sreejan Kumar (sreejank@princeton.edu). Data and code repository available at:
https://osf.io/srmn3/overview?view_only=1d55cecc30ef4b9b9156267480c83c7d

Abstract

Determining the degree of similarity between different stimuli is central for intelligent behavior. Psychologists have long debated the nature of similarity, considering whether it is continuous and space-like or discrete and feature-like. Here, we argue that the idea that stimuli are similar to the extent that they are likely to have been generated by the same process unifies these spatial and featural accounts as special cases. We formulate this notion of generative similarity in terms of Bayesian inference over hierarchical generative processes and show how it can illuminate key ideas from the literature including the universal law of generalization, feature contrast models, exemplar and prototype theories, and metric violations in semantic organization. Moreover, just as previous work on similarity drew on connections to multi-dimensional scaling and additive clustering, generative similarity provides a new way to construct representations of stimuli by drawing on contrastive learning, a modern machine learning procedure for learning representations by pushing similar stimuli together and dissimilar stimuli apart. We show how this approach can provide a way to translate hierarchical generative processes into spatial representations that support human-like abstraction and generalization, even in complex domains that are parameterized by probabilistic programs.

Keywords: Similarity, Generative Processes, Bayesian Inference, Representations

Generative Similarity, Generalization, and Abstraction

Faced with complex, ever-changing sensory environments, humans must decide whether knowledge generalizes from one experience to another, and whether such experiences can be viewed as instances of the same abstract type. These problems of generalization and abstraction are at the core of human intelligence and our ability to learn from small amounts of data (Griffiths, 2020; Shepard, 1987). Key to tackling these problems is the ability to reason about what makes different stimuli similar, an ability that has been at the center of extensive modeling efforts over the past decades (Chater & Vitányi, 2003; Destler et al., 2023; Kemp et al., 2005; Medin et al., 1993; Myung & Shepard, 1996; Shepard, 1987; Sims, 2018; Tenenbaum & Griffiths, 2001; Tversky, 1977).

Different models of similarity have been proposed in the literature, foremost of which are the spatial model of Shepard (1987), and the feature model of Tversky (1977). Shepard (1987) formulated similarity as distance in psychological space, which captures the extent to which pairs of stimuli fall within the same “consequential region” for some property (e.g., having an enzyme or being poisonous). Tversky (1977) argued that similarity may be better described as a set-theoretic measure applied to stimuli that are represented as discrete sets of features. Tenenbaum and Griffiths (2001) unified the former accounts by showing that they correspond to different ways of specifying hypotheses (continuous vs. discrete) for a Bayesian inference over whether a given property that holds for one stimulus also holds for another.

These accounts of similarity tend to emphasize the apparent dichotomy between “features” and “regions”. In fact, these are just two implicit ways of specifying an underlying generative distribution of the data. We will argue that it is these underlying generative processes that really matter from the point of view of the agent’s generalization and abstraction behavior, proposing an alternative notion of similarity based on Bayesian inference over generative processes (an initial version of this idea was proposed in Kemp et al. (2005), and more recently it has been explored as a measure of shape similarity in

Destler et al., 2023). Given a hierarchical generative model of the environment whereby data distributions are sampled and then individual stimuli are generated from those distributions, we define the similarity of a pair of stimuli as the probability that those stimuli were generated from the same underlying distribution. We demonstrate the unifying power of this approach by showing how different classic results from the literature, including the universal law of generalization (Shepard, 1987), Tversky’s feature contrast model (Tversky, 1977), exemplar and prototype theories (Battleday et al., 2020; Medin & Schaffer, 1978; Posner & Keele, 1968), and metric violations in semantic organization (Griffiths et al., 2007; Tversky & Hutchinson, 1986) correspond to specific choices of generative models.

Moreover, just as spatial models of similarity build on multi-dimensional scaling (Shepard, 1980), and featural models of similarity build on additive clustering (Shepard & Arabie, 1979), we show how generative similarity can be combined with contrastive learning (Chen et al., 2020), a framework for representation learning from the machine learning literature, to derive spatial representations from generative models that support human-like generalization and abstraction. In particular, we show how they can be used to predict human judgments across various domains, including geometric reasoning, natural images, and abstract shapes parametrized by probabilistic programs that are neither feature-like nor space-like.

In the remainder of the paper, we first review previous accounts of similarity. We then introduce the generative similarity formalism and its theoretical implications. Next, we introduce the generative-contrastive framework for deriving representations and apply it to various empirical domains. Finally, we conclude with a discussion of the implications of our work for the psychological literature.

Models of Similarity

At the core of any model of similarity is the question of how an agent should decide on the extent to which two stimuli are similar. Different answers to this question lead to

different models of similarity. In spatial models of similarity, as exemplified by the work of Shepard (1987), the similarity between stimuli is taken to be inversely related to their distance in *psychological* (or representational) space. More specifically, Shepard (1987) (and later Tenenbaum and Griffiths, 2001), formulated similarity between two stimuli x and y as a response to the query “how likely is it that y possesses a certain property c having observed x possessing it too?”. The possession of a property is operationalized as the stimulus being within some latent *consequential region* h in psychological space. The probability of possessing a shared property is then calculated by summing over all shared consequential regions. The end result of this procedure is the *universal law of generalization* (Shepard, 1987) which states that similarity between two stimuli should decay as a convex function of their psychological distance, typically approximated as an exponential

$$s(x, y) \propto e^{-d(x, y)} \quad (1)$$

where $d(x, y)$ is psychological distance. This idea has been elaborated on both empirically (e.g., Cheng, 2000; Marjeh et al., 2024), and theoretically using tools from rate-distortion theory (Sims, 2018), maximum entropy estimation (Myung & Shepard, 1996), and algorithmic transformational distance (Chater & Vitányi, 2003). It also serves as a backbone for exemplar and prototype theories of categorization (Battleday et al., 2020; McKinley & Nosofsky, 1995; Nosofsky, 1986; Reed, 1972).

One classic criticism of spatial models of similarity is their commitment to the metric axioms (Tversky, 1977; Tversky and Hutchinson, 1986), namely that spatial distances are symmetric and satisfy the triangle inequality (though see Krumhansl (1978), Nosofsky (1992), and Jones et al. (2018) for possible resolutions). Tversky (1977) noted that human similarity judgments can violate these properties in certain cases such as in semantic domains (e.g., the concept *gas-lantern* is similar to *moon*, the concept *moon* is similar to *ball*, but *gas-lantern* and *ball* are not similar; James, 1890). Using this insight, Tversky argued that similarity is better captured by an alternative class of models based on set-theoretic comparisons between stimulus features. Demanding that any such

112 similarity measure should satisfy a set of “common-sense” axioms (Tversky, 1977) he then
 113 narrowed down its form to the following formula

$$s_{ij} = \alpha_s f(\Sigma_i \cap \Sigma_j) - \beta_s f(\Sigma_i - \Sigma_j) - \gamma_s f(\Sigma_j - \Sigma_i) \quad (2)$$

114 for some non-negative numbers $\alpha_s, \beta_s, \gamma_s$, and a non-negative scalar function f (e.g., set
 115 size $f(A) = |A|$) where Σ_i and Σ_j are the sets of features possessed by stimuli σ_i and σ_j ,
 116 and $A \cap B$ and $A - B$ correspond to the sets of features that are in A and B , and in A but
 117 not in B , respectively. For different choices of parameters, one can account for phenomena
 118 such as asymmetry in similarity judgments (e.g., when $\beta_s \neq \gamma_s > 0$). Moreover, as with
 119 Shepard’s approach, more general formulations have been proposed following Tversky’s
 120 work (e.g., using fuzzy logic; Santini and Jain, 1999).

121 Moving beyond the spatial and featural models, Tenenbaum and Griffiths (2001)
 122 observed that Shepard and Tversky’s accounts can be subsumed by a probabilistic
 123 formulation of similarity using Bayesian inference. Specifically, let C denote the (latent)
 124 consequential region that leads to the possession of a property of interest. Similarity
 125 between two stimuli x, y is then defined as the probability that $y \in C$ having observed x
 126 (which possesses the property of interest and hence is also in C). This is determined by the
 127 probability that C corresponds to a hypothesized region h given the observation of x ,
 128 $p(C = h|x)$, which is computed using Bayes rule, and summing over all possible h .
 129 Crucially, Tenenbaum and Griffiths (2001) noted that the Bayesian formulation is agnostic
 130 to the specification of h which could equally be a discrete consequential subset determined
 131 by features rather than a continuous consequential region. By considering both options,
 132 the authors showed that the universal law and Tversky’s feature contrast model can be
 133 accommodated.

134 The existing Bayesian treatment of similarity as generalization has two limitations:
 135 (i) it is not always clear which property humans are reasoning about when they provide a
 136 similarity judgment, and, in fact, there may be no single or definite answer, and (ii) the
 137 distinction between continuous and discrete hypothesis spaces is merely mathematical and

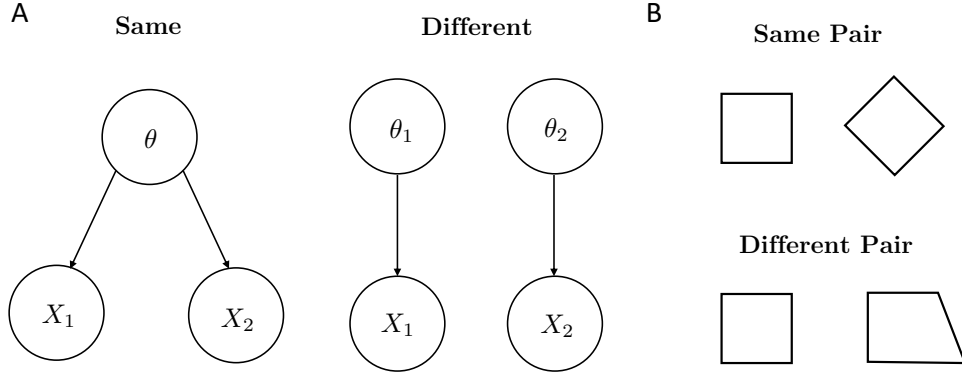


Figure 1

*Schematic representation of generative similarity. **A.** Graphical models for the same and different data generation hypotheses. **B.** Example same and different stimulus pairs.*

does not clearly specify the agent’s model of the stimulus domain. To address these limitations, an alternative notion of Bayesian similarity has been proposed in the literature based on an inference about the process that might have generated the observed stimuli (Destler et al., 2023; Kemp et al., 2005). This generative approach asks the question “how likely are the stimuli to have been generated by the same process?”. However, this notion of similarity has not been extensively studied, leaving open questions about its ability to capture the universal law of generalization and other foundational results, as well as how it can be estimated and used in practice. Our work aims to address this gap (i) by directly showing how generative similarity, when combined with a suitable generative model, can capture prior theoretical findings, and (ii) by combining generative similarity with the method of contrastive learning developed in the machine learning literature to learn representations that can be used to predict human judgments, even when it could not otherwise be calculated.

Defining Generative Similarity

We begin by specifying the probabilistic formulation of generative similarity. Consider an agent whose model of a certain stimulus domain (e.g., geometric shapes) is given by a hierarchical generative process. In the first part of this process, distributions are

sampled from some prior $p(\theta)$ – for example, θ could specify a distribution over visual attributes that define a certain visual category (e.g., a rectangle). Given the chosen parameters, stimuli are then sampled from the selected data distribution $p(d|\theta)$ (e.g., a specific image of a rectangle is produced). Under this model, the data distribution is then given by summing over all possible values of θ weighted by the prior, namely,

$$p(d) = \int d\theta p(d|\theta)p(\theta).$$

Recall that our goal is to quantify the extent to which two stimuli x, y are likely to have been generated from the same underlying distribution, or their *generative similarity* $s_{gen}(x, y)$. A natural way to do so is through the Bayes odds ratio which compares the probability that x, y have been sampled from the *same* distribution relative to that of them being sampled from *different* (i.e., independent) distributions (Figure 1)

$$s_{gen}(x, y) = \frac{p(\text{same}|x, y)}{p(\text{different}|x, y)}. \quad (3)$$

Now, we can expand each term above using Bayes rule $p(h|x, y) = p(x, y|h)p(h)/p(x, y)$ where h is any hypothesis, and since we are interested in the ratio between the posteriors of two hypotheses, the normalization factor $p(x, y)$ cancels out. From here, based on the definitions of the “same” and “different” (diff. for short) processes (Figure 1A) we have

$$p(\text{same}|x, y) \propto p(x, y|\text{same})p(\text{same}) = \int p(x|\theta)p(y|\theta)p(\theta)d\theta \times p(\text{same}) \quad (4)$$

and likewise,

$$p(\text{diff.}|x, y) \propto p(x, y|\text{diff.})p(\text{diff.}) = \int p(x|\theta_1)p(y|\theta_2)p(\theta_1)p(\theta_2)d\theta_1d\theta_2 \times p(\text{diff.}), \quad (5)$$

which capture the probability that x and y are generated from a single distribution (with random parameter θ) vs. two independently-sampled distributions (with random parameters θ_1, θ_2), respectively. Note that $p(\text{same})$ and $p(\text{diff.})$ are the prior probabilities for the hypotheses that stimuli are sampled from same and different distributions in general. We will take these to be equal a priori, and hence they cancel out in the ratio

(alternatively, they simply contribute a constant scale that is independent of the data).

Thus, we arrive at the final form of generative similarity

$$s_{gen}(x, y) = \frac{\int p(x|\theta)p(y|\theta)p(\theta)d\theta}{\int p(x|\theta_1)p(y|\theta_2)p(\theta_1)p(\theta_2)d\theta_1d\theta_2}. \quad (6)$$

In summary, generative similarity provides a natural way to quantify the similarity between a pair of stimuli given their probabilities under a suitable generative model. This is done by comparing the probability of the pair under a case in which one distribution is sampled and the two stimuli are generated from it, relative to that of the case where two independent distributions are sampled and then each stimulus is generated separately from their respective distributions.

Generative Similarity as a Unifying Framework

Our goal next is to show how generative similarity can unify aspects of psychological similarity that are central to the literature reviewed earlier, namely, (i) the universal law of generalization (Shepard, 1987), (ii) the feature contrast model (Tversky, 1977), (iii) prototype and exemplar theories of categorization (Nosofsky, 1986; Reed, 1972), and (iv) metric violations in semantic organization (Griffiths et al., 2007; Tversky, 1977; Tversky & Hutchinson, 1986).

The Universal Law of Generalization

We begin by considering how generative similarity connects to Shepard’s universal law of generalization in its Bayesian form (Tenenbaum & Griffiths, 2001). Recall that the formal setting of Shepard’s law is that of a rational agent that considers whether a certain property generalizes from one stimulus x to another y by reasoning about the consequential regions these stimuli belong to in psychological space. The task of the rational agent is to ask “which consequential region is x likely to be associated with, and does y fall within that region too?”. Being “associated” with a consequential region is typically interpreted in one of two ways (Tenenbaum & Griffiths, 2001): i) *Weak sampling*: the stimulus x is sampled independently from the consequential region h and is associated with it by

201 happening to fall within it, and ii) *Strong sampling*: the stimulus x is associated with a
 202 consequential region by being directly (and uniformly) sampled from it. As argued by
 203 Tenenbaum and Griffiths (2001), the strong sampling assumption carries with it
 204 explanatory power when generalizing Shepard’s account to more complex setups (such as
 205 generalizing from multiple examples) that are not captured well by weak sampling.

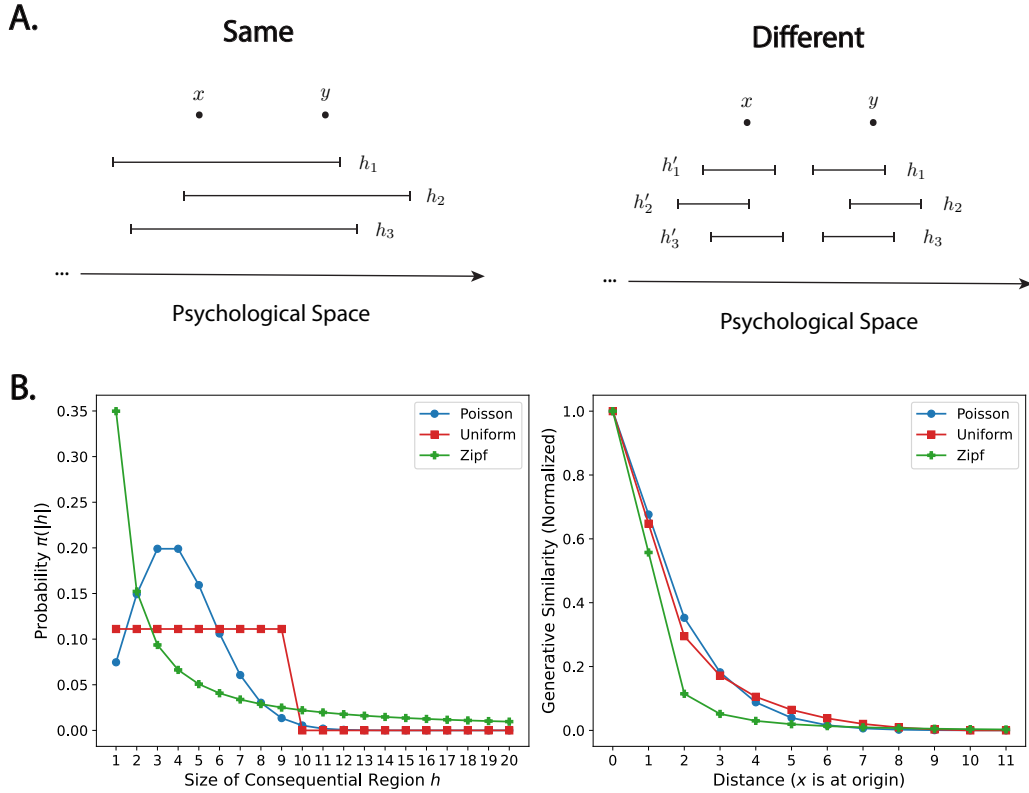


Figure 2

Generative account of the universal law of generalization. A. The “same” and “different” generative processes in which consequential regions h are sampled, and then points x and y are sampled from them. *B.* Simulation results showing that the convexity of the derived generative similarity is insensitive to different choices of the prior $\pi(|h|)$ for three different distributions: a Poisson distribution, a heavy-tailed Zipf distribution, and a uniform distribution on sizes up to 9.

directly and uniformly from consequential regions) for two reasons: (i) due to its role in extending Shepard’s formulation to multiple items (Tenenbaum, 1999; Tenenbaum & Griffiths, 2001), and (ii) because it provides a clear specification of a generative account: reasoning about whether properties generalize from one stimulus to another becomes that of reasoning about the consequential regions that generated those stimuli. Put differently, stimuli are likely to possess the same property if they are likely to have been sampled from the same consequential region. Thus, the underlying generative process that the rational agent is reasoning about is that of first sampling a consequential region, and then sampling points from it (Figure 2A). Our goal in what follows is to delineate how generative similarity under this specific generative process relates to the universal law and the previous Bayesian formulation of Tenenbaum and Griffiths (2001).

Starting from the formulation of Tenenbaum and Griffiths (2001), let C denote the latent consequential region that leads to possessing a property of interest. The Bayesian computation then proceeds by evaluating the probability that $y \in C$ having observed x (which possesses that property)

$$\begin{aligned}
 s_g(x, y) &= p(y \in C | x) \\
 &= \sum_h p(y \in C | C = h, x) p(C = h | x) \\
 &= \sum_h p(y \in C | C = h) p(C = h | x) \\
 &= \sum_{h: y \in h} p(C = h | x)
 \end{aligned}$$

where the third and fourth equalities follow from the fact that knowing C , y becomes independent of x , and $p(y \in C | C = h)$ becomes a deterministic indicator of whether y is in C or not. Next, from strong sampling, i.e., that x was sampled from C , we have

$$p(x | C = h) = \begin{cases} \frac{1}{|h|}, & x \in h \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

225 which when combined with Bayes rule $p(h|x) = p(x|h)\pi(h)/\sum_{h'} p(x|h')\pi(h')$ yields

$$s_g(x, y) = \frac{\sum_{h:x,y \in h} \frac{1}{|h|} \pi(h)}{\sum_{h':x \in h'} \frac{1}{|h'|} \pi(h')}. \quad (8)$$

226 where $\pi(h)$ is the prior over consequential regions. Following Shepard (1987) and
 227 Tenenbaum and Griffiths (2001), we make the following assumptions about the nature of
 228 consequential regions: (i) the prior over consequential regions depends only on the size of h
 229 but not on its location, i.e., $\pi(h) = \pi(|h|)$, (ii) the consequential regions are connected and
 230 convex¹ (i.e., if $x, y \in h$ then the line connecting them is also in h), (iii) there are no special
 231 points in the space and so they can be treated generically by setting one of them at the
 232 origin, namely, x , and considering different y values. Given these assumptions, the authors
 233 argued that the shape of s_g is monotonically decreasing and convex as a function of the
 234 separation between x and y , irrespective of the choice of the prior. This yields the universal
 235 law of generalization (for a proof see Appendix A). Note also that these assumptions
 236 further imply that the denominator in (8) is a constant independent of x (since we are
 237 summing over all the consequential regions containing a single generic point), and so we
 238 can write $\sum_{h':x \in h'} \pi(|h'|)/|h'| \equiv 1/s_0$ from which it follows that

$$s_g(x, y) = s_0 \sum_{h:x,y \in h} \frac{1}{|h|} \pi(|h|). \quad (9)$$

239 Turning next to generative similarity, under the generative model shown in Figure
 240 2A (i.e., a consequential region is sampled and then points are sampled uniformly from it)
 241 it is given by

$$s_{gen}(x, y) = \frac{\sum_h p(x|h)p(y|h)\pi(h)}{\sum_{h_1} p(x|h_1)\pi(h_1) \sum_{h_2} p(y|h_2)\pi(h_2)}. \quad (10)$$

242 This expression further simplifies under the same assumptions regarding consequential
 243 regions and strong sampling (7) noted above, namely,

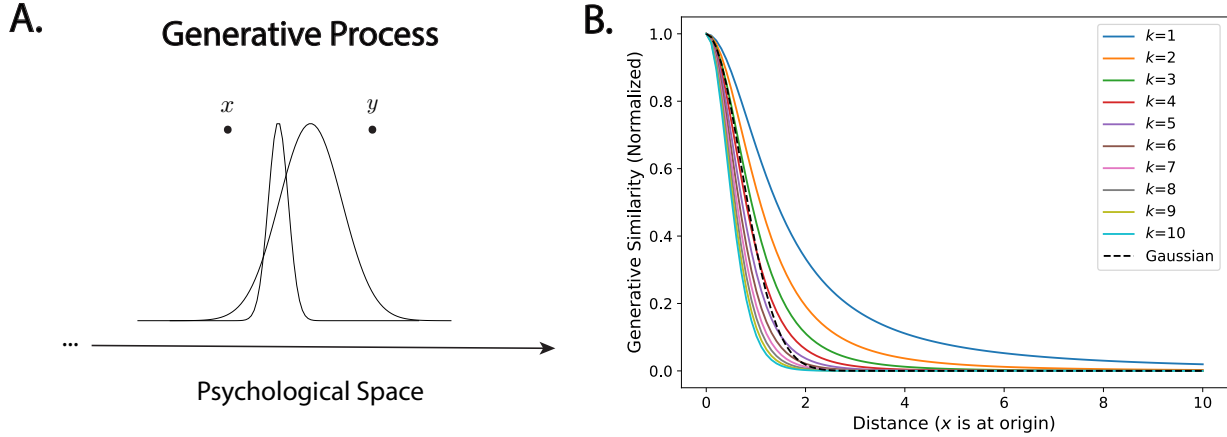
$$s_{gen}(x, y) = s_0^2 \sum_{h:x,y \in h} \frac{1}{|h|^2} \pi(|h|). \quad (11)$$

¹ It is worth noting that Shepard (1987) suggested that this property may be relaxed to centrally symmetric regions which may or may not be convex.

We see that s_{gen} and s_g essentially capture the same function up to a different weighting of the size of consequential regions. Crucially, this difference has no effect on the convexity of the similarity measure as a function of distance between x and y (see Appendix A for a proof). It does imply, however, that s_{gen} decays faster than s_g which is expected as the former concerns reasoning about the the sampling of two points from the consequential region as opposed to one. This is compatible with the faster decay rate discussed by Tenenbaum and Griffiths (2001) when dealing with generalization from multiple examples. As a sanity check, we confirmed in a simulation that Equation (10) indeed yields convex curves for various choices of the prior distribution (Figure 2B). Thus, we conclude that generative similarity can recover Shepard’s law of generalization under the same assumptions used in prior work.

A classic debate in the literature on the universal law concerns the existence of a small violation of convexity for sufficiently close points in psychological space (Chater & Vitányi, 2003; Nosofsky, 1985). This violation of convexity is often approximated as a type of Gaussian generalization gradient that is employed in exemplar-based theories of categorization (Medin & Schaffer, 1978; Nosofsky, 1986). It has also been argued to arise from the inherent stochasticity of representations, whereby individual stimulus representations are sampled from some noisy Gaussian distributions (Ennis et al., 1988; Nosofsky, 1992). As we show next, generative similarity can provide a way to account for this variant by tying it to a different hypothesis regarding the generative process underlying the data. Rather than reasoning about whether stimuli were sampled from the same consequential region via strong sampling, we will consider whether they were generated from the same underlying Gaussian distribution (Figure 3A).

To that end, consider a data generative process in which D -dimensional Gaussian distributions are sampled, and then stimuli are generated from those distributions (Figure 3A). Specifically, we assume that the parameters of a Gaussian distribution, namely, its mean μ and covariance Σ , are sampled from some priors $p(\mu)$ and $p(\Sigma)$, after which data


Figure 3

Capturing other variants of the universal law. **A.** Indefinite Gaussian generative process in which Gaussian distributions are sampled and then points are sampled from those distributions. **B.** Corresponding one-dimensional generative similarity as a function of distance from a reference point $x = 0$ at the origin for different inverse Wishart prior hyperparameters k . Dashed line indicates a Gaussian gradient e^{-x^2} for reference.

points are sampled from that Gaussian $X \sim \mathcal{N}(\mu, \Sigma)$. Following Griffiths and Tenenbaum (2007), we will consider a uniform prior over the means for some large region R , and an inverse Wishart prior (a common conjugate prior on covariance matrices; see e.g., Gelman et al., 1995; Minka, 1998) with hyperparameters $k \geq D$ and I where I is the identity matrix. Formally, we have

$$p(\mu) = \frac{1}{|R|}, \quad p(\Sigma) = \frac{1}{c_{kD} |\Sigma|^{(k+D+1)/2}} \exp \left[-\frac{1}{2} \text{tr}(\Sigma^{-1}) \right] \quad (12)$$

where $c_{kD} > 0$ is a constant, and $|\cdot|$ and $\text{tr}(\cdot)$ are the determinant and trace operators, respectively (with the exception of $|R|$ which corresponds to the size of the region R). Under these priors, it is possible to derive the probability of a set of data points $d = \{x_1, \dots, x_n\}$ being sampled from a single *indefinite* Gaussian (for a derivation, see Appendix B or Griffiths and Tenenbaum, 2007)

$$p(d|\text{Gaussian}) \propto \frac{1}{|S + I|^{(n+k-1)/2}} \quad (13)$$

where $S = \sum_{i=1}^n (x_i - \bar{x})(x_i - \bar{x})^\top$ is the scatter matrix associated with the data,
 $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$ is the data mean, and I is the identity matrix as before. Note that for a
 single data point, we have $S = 0$ which implies that Equation (13) is constant in that case.
 This is sensible as any individual point is equally likely to be sampled from *some* Gaussian.
 Thus, when comparing the generative similarity between two points x, y (i.e., the ratio
 between their probability of being sampled from the same Gaussian vs. two independent
 Gaussians) we have

$$s(x, y) \propto p(x, y | \text{Gaussian}) \propto \frac{1}{|S + I|^{(1+k)/2}}. \quad (14)$$

To make this concrete, for $D = 1$ and setting $x = 0$ we have $s(0, y) = 1/(1 + y^2/2)^{(1+k)/2}$.
 As shown in Figure 3B, this function is monotonically decreasing and is largely convex
 with the exception of the area near the origin.

Thus, we see that generative similarity provides the right level of generality to
 meaningfully unify the universal law and its variant by tying each to specific choices of
 generative processes. As we will show later, the generic Gaussian generative process
 presented here will allow generative similarity to make direct contact with the well-known
 exemplar and prototype theories of categorization, but before doing so we will consider
 another competing account of similarity, namely, Tversky’s feature model (Tversky, 1977).

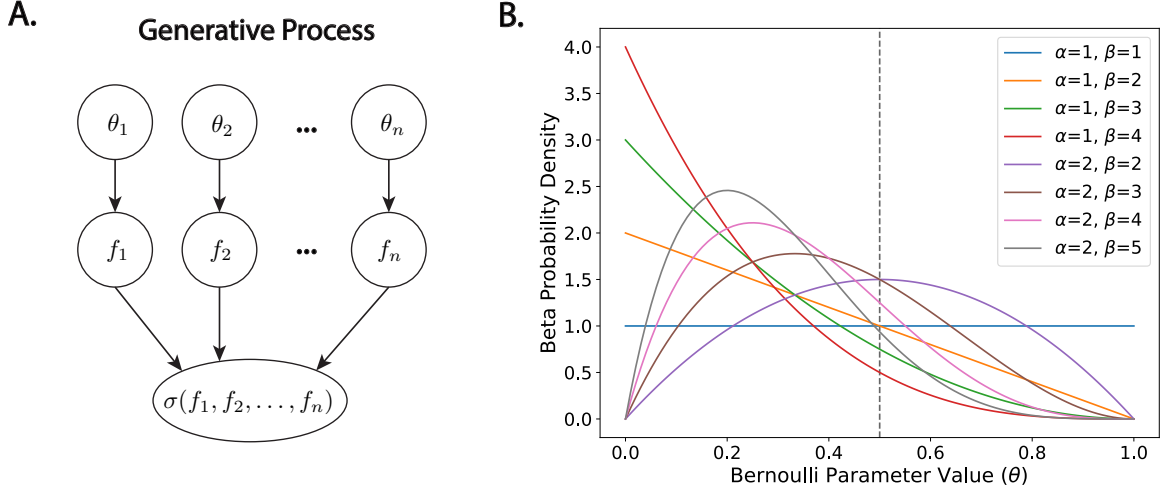
Feature Contrast Models and Diagnosticity

Our goal in what follows is to show how generative similarity can capture Tversky’s
 feature-based model of similarity (Tversky, 1977). Recall that under this model, the
 similarity between a pair of stimuli σ_1 and σ_2 with feature sets² Σ_1 and Σ_2 , respectively, is
 given by the formula

$$s = \alpha_s f(\Sigma_1 \cap \Sigma_2) - \beta_s f(\Sigma_1 - \Sigma_2) - \gamma_s f(\Sigma_2 - \Sigma_1) \quad (15)$$

for some non-negative numbers $\alpha_s, \beta_s, \gamma_s$, and some non-negative scalar function (often
 taken to be set size $f(A) = |A|$). As in the previous section, generative similarity requires

² These should not be confused with the covariance matrix notation of the previous section.


Figure 4

Feature sampling and Tversky's contrast model. A. Beta-Bernoulli generative process associated with feature-based stimuli. B. Example Beta distributions under the constraint $\alpha \leq \beta$. Dashed line indicates the 1/2 threshold.

specifying a suitable generative model of the data. In our case, this amounts to a process that generates features, and then generates stimuli that possess those features (Figure 4A). This can be achieved using Beta-Bernoulli conjugate distributions (a variant of this process was also explored in Kemp et al., 2005). Specifically, given a set of available binary features $i = 1, \dots, n$ (e.g., “is symmetric”, “has four edges”, etc.), we sample Bernoulli parameters θ_i for each feature i from a prior $\theta_i \sim \text{Beta}(\alpha, \beta)$, and then sample a corresponding Bernoulli indicator $f_i \sim \text{Bern}(\theta_i)$ where $f_i = 1$ if the property holds true (with probability θ_i) and 0 otherwise. Finally, given the feature vector $f \equiv \{f_i\}$ we uniformly sample a stimulus $\sigma(f)$ that possesses those features from the set of all such stimuli $\mathcal{S}(f)$. Mathematically, we have

$$(\sigma, \{f_i\}, \{\theta_i\}) = \begin{cases} \theta_i \sim \text{Beta}(\alpha, \beta), & \text{Sample Bernoulli parameters} \\ f_i \sim \text{Bern}(\theta_i), & \text{Sample discrete features} \\ \sigma(f) \sim \text{Uniform}(\mathcal{S}(f)), & \text{Sample example stimulus.} \end{cases} \quad (16)$$

In Appendix C, we show that generative similarity under this process can be computed in

closed form and is given by

$$\log s(\sigma_1, \sigma_2) = |\Sigma_1 \cap \Sigma_2| \log \frac{\alpha + 1}{\alpha} \frac{\beta}{\beta + 1} - (|\Sigma_1 - \Sigma_2| + |\Sigma_2 - \Sigma_1|) \log \frac{\beta + 1}{\beta} + \log c_0 \quad (17)$$

where $c_0 > 0$ is a constant independent of the stimuli. From here we see that generative similarity captures Tversky’s proposed formula (up to a monotonically increasing transformation $s_{\text{gen}} = c_0 e^{s_{\text{Tversky}}}$) with the parameter values

$$\alpha_s = \log \frac{\alpha + 1}{\alpha} \frac{\beta}{\beta + 1}, \quad \beta_s = \gamma_s = \log \frac{\beta + 1}{\beta}. \quad (18)$$

Interestingly, while β_s and γ_s are always positive in this case, to ensure that α_s is non-negative we must require that $\beta \geq \alpha$ which corresponds to a Beta distribution with mean value $\alpha/(\alpha + \beta) \leq 1/2$ (see Figure 4B for examples). This reflects a rational agent who assumes that features are typically not realized in stimuli and therefore are *diagnostic*. This is because on average $\theta_i \leq 1/2$, which means that a property is more likely not to hold than to hold (see also Oaksford and Chater (1994) for related ideas). Moreover, observe that the condition $\beta_s = \gamma_s$ implies that the similarity formula is symmetric, which is a natural outcome of our original definition of generative similarity (the probability of stimuli being generated from the same process does not take into account directionality). Nevertheless, later on we will show how asymmetric similarity can be naturally incorporated within the generative framework.

Beyond the connection to Tversky’s formulation, it is worth noting that Equation (17) can also be written as (see Appendix C)

$$\log s(\sigma_1, \sigma_2) = |\Sigma_1 \cap \Sigma_2| \log \frac{\alpha + 1}{\alpha} - |\Sigma_1 \cup \Sigma_2| \log \frac{\beta + 1}{\beta} + \log c_0. \quad (19)$$

where $|\Sigma_1 \cup \Sigma_2|$ is the union of all features that hold true either in σ_1 or σ_2 . This provides another intuitive interpretation: generative similarity between a pair of stimuli tracks the number of shared features relative to the overall number of features possessed by those stimuli, weighted by the parameters of the Beta distribution.

Generative Similarity and Categorization

Generative similarity provides a clear avenue for reasoning about categories: just as we can ask how likely a pair of stimuli x_0 and x_1 are to have been generated from the same distribution, we can also ask how likely a stimulus x_0 and a set of stimuli $\{x_i\}$ are to have been generated from the same distribution. This formulation is analogous to an exemplar-based theory of categorization (Nosofsky, 1986) whereby category membership is determined based on the similarity between a stimulus and instances of the category in question (i.e., its exemplars). Crucially, our approach directly links this notion of similarity to inference with respect to the underlying distribution that specifies the category. To illustrate this point, we shall consider again the Gaussian sampling generative process defined in Equation (12). Recall that under that process, the probability of a set of stimuli $d = \{x_1, \dots, x_n\}$ (or $\{x_i\}$ in short), given that they were sampled from the same indefinite Gaussian, is given by

$$p_G(d) \equiv p(d|\text{Gaussian}) \propto \frac{1}{|S + I|^{(n+k-1)/2}} \quad (20)$$

where $S(\{x_i\}) = \sum_{i=1}^n (x_i - \bar{x})(x_i - \bar{x})^\top$ is the scatter matrix associated with the data, $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$ is the data mean, I is the identity matrix, and $|\cdot|$ is the determinant. Now, applying Equation (20) to the definition of generative similarity (6) we have

$$s(\{x_i\}, x_0) = \frac{p_G(\{x_i\} \cup \{x_0\})}{p_G(\{x_i\})p_G(\{x_0\})} \propto \frac{|S(\{x_i\}) + I|^{(n+k-1)/2}}{|S(\{x_i\} \cup \{x_0\}) + I|^{(n+k)/2}} \quad (21)$$

where, as before, we used the fact that for a single data point p_G is constant. Now, to see how this formula relates to familiar models of categorization, we will consider the limit $n \rightarrow \infty$, i.e., when the number of category exemplars is large. In Appendix D, we show that in this limit Equation (21) assumes the following familiar form

$$s(\{x_i\}, x_0) \rightarrow \frac{1}{\sqrt{(2\pi)^D |\Sigma(\{x_i\})|}} \exp \left(-\frac{1}{2} (x_0 - \bar{x}(\{x_i\}))^\top \Sigma^{-1}(\{x_i\}) (x_0 - \bar{x}(\{x_i\})) \right) \quad (22)$$

where $\bar{x}(\{x_i\}) = \frac{1}{n} \sum_{i=1}^n x_i$ and $\Sigma(\{x_i\}) = \frac{1}{n} S(\{x_i\})$ are the empirical mean and covariance of the data $\{x_i\}$, respectively (we also confirmed this result in a simulation as can be seen

in Supplementary Figure D1). This means that generative similarity between a point and a multidimensional exemplar set converges to the probability of that point under a Gaussian distribution specified by the empirical mean $\bar{x}(\{x_i\})$ and covariance $\Sigma(\{x_i\})$ of that set. Equivalently, it is a monotonically decreasing function of the *Mahalanobis distance* (Mahalanobis, 2018) between x_0 and the mean “prototype” $\bar{x}(\{x_i\})$, weighted by the covariance matrix $\Sigma(\{x_i\})$. Note also that this highlights generative similarity’s pressure towards compressed representations (here the mean and covariance of the data) by directly reasoning about the underlying distribution that generated the data.

Semantic Organization and Metric Axiom Violations

A complete theory of similarity must be able to capture violations of metric axioms that challenge spatial models in semantic domains (Griffiths et al., 2007; Tversky, 1977; Tversky & Hutchinson, 1986). Our goal next is to show how generative similarity can account for such phenomena.

Symmetry Violations

The current definition of generative similarity assumes a symmetric form as it responds to the query “how likely is it that x and y were generated from the same distribution relative to being generated independently?”. To see how generative similarity can nevertheless capture asymmetries, observe that the previous query is probabilistically equivalent to the following one: “how likely is it that x was generated from the same distribution that generated y relative to being generated independently?”. Indeed, in that case we have

$$s(x, y) = \frac{\int p(x|\theta)p(\theta|y)d\theta}{\int p(x|\theta)p(\theta)d\theta} \quad (23)$$

and using Bayes rule to expand $p(\theta|y)$ we have

$$s(x, y) = \frac{\int p(x|\theta)p(y|\theta)p(\theta)d\theta}{\int p(x|\theta)p(\theta)d\theta \int p(y|\theta)p(\theta)d\theta} \quad (24)$$

which is the original (symmetric) definition. Crucially, note that the symmetry emerges due to the division by the independent sampling baseline $\int p(x|\theta)p(\theta)d\theta$ (i.e., the probability or

frequency of the stimulus x under the generative model). This gives us a clear way to accommodate for asymmetries by simply dropping the normalization against the independent baseline. In that case, the asymmetric generative similarity of y to x (i.e., if y is sampled from some distribution, how likely is x to be sampled from it too?)³ becomes

$$s(y; x) = \int p(x|\theta)p(\theta|y)d\theta = \frac{\int p(x|\theta)p(y|\theta)p(\theta)d\theta}{\int p(y|\theta)p(\theta)d\theta} \quad (25)$$

which is indeed asymmetric (we use a semicolon to highlight the asymmetry and distinguish it from the symmetric definition). Importantly, this is precisely the frequency-based structure that generates asymmetries in other probabilistic frameworks such as topic models (Griffiths et al., 2007) which successfully capture metric violations in semantic judgments. Indeed, taking the ratio of the two-way similarities we have

$$\frac{s(y; x)}{s(x; y)} = \frac{\int p(x|\theta)p(\theta)d\theta}{\int p(y|\theta)p(\theta)d\theta} = \frac{p(x)}{p(y)} \quad (26)$$

which suggests that if $p(y) \ll p(x)$ (e.g., if y is highly associated with only a small subset of θ values) then $s(y; x) \gg s(x; y)$, i.e. the similarity of y to x is bigger than that of x to y (assuming that their joint sampling probability $\int p(x|\theta)p(y|\theta)p(\theta)d\theta$ is non-vanishing). To make this concrete, and to highlight the analogy with topic models, consider the word pair *cobra-snake* (Griffiths et al., 2007): The word *cobra* is more specialized than the word *snake* and, correspondingly, has lower frequency $p(\text{cobra}) \ll p(\text{snake})$. This in turn implies that $s(\text{cobra}; \text{snake}) \gg s(\text{snake}; \text{cobra})$ which is what one observes in the literature (Griffiths et al., 2007): the similarity (or association) of *cobra* to *snake* is larger than that of *snake* to *cobra*. Equivalently, the distribution (or ‘topic’) that generated the word *cobra* is likely to generate the word *snake*, but the distribution (or ‘topic’) that generated the word *snake* might not necessarily be focused on cobras and so is not as likely to generate the word *cobra*. In summary, frequency-dependent asymmetries in similarity judgment can

³ Another intuitive (though less formal) way to think of this definition is: if y is associated with a given distribution (e.g., a category or topic), how likely is x to be associated with it too?

be easily accounted for in the generative similarity framework by simply dropping the comparison against the independent sampling baseline.

Triangle Inequality Violations

The triangle inequality states that any spatial distance metric $d(\cdot, \cdot)$ must satisfy $d(x, z) \leq d(x, y) + d(y, z)$ for all x, y, z . This implies that any spatial measure of similarity that is monotonically decreasing in $d(\cdot, \cdot)$ is constrained by the triangle inequality. In particular, the triangle inequality is violated for stimulus triplets x, y, z for which $s(x, y)$ is high, $s(y, z)$ is high, but $s(x, z)$ is low. This typically happens when the high similarity pairs are similar in different respects. Well-known examples of such violations are found in the domain of analogies (Peterson et al., 2020): Consider the conceptual relations *nurse-patient*, *mother-baby*, and *frog-tadpole*. The relation *nurse-patient* is similar to *mother-baby* (with respect to care), the relation *mother-baby* is similar to *frog-tadpole* (with respect to birth), however, *nurse-patient* and *frog-tadpole* are not similar.

Unlike spatial models of similarity, generative similarity is not constrained by the triangle inequality. To demonstrate this, consider a hierarchical generative process in which one of two distributions $\phi_1(x), \phi_2(x)$ is chosen at random (e.g., each corresponding to a different class of analogical relations) and then data are sampled from the selected distribution. Crucially, let us assume that for a certain triplet (x, y, z) the distributions satisfy $\phi_1(x) = \phi_1(y) = c$ for some finite constant c but $\phi_1(z) = \epsilon \ll c$ (i.e., x, y but not z have high probability under ϕ_1). Likewise, $\phi_2(y) = \phi_2(z) = c$ but $\phi_2(x) = \epsilon \ll c$ (Figure 5A). In this case, computing generative similarity based on Equation (6) yields

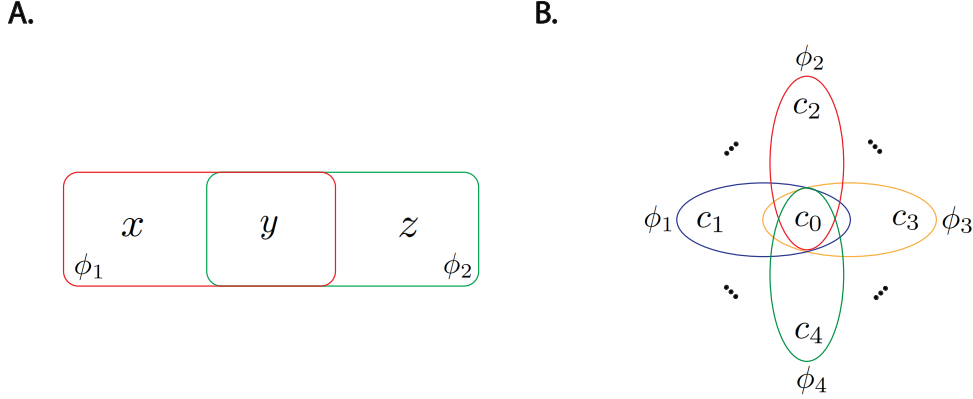
$$s(x, y) = \frac{\phi_1(x)\phi_1(y)/2 + \phi_2(x)\phi_2(y)/2}{(\phi_1(x)/2 + \phi_2(x)/2)(\phi_1(y)/2 + \phi_2(y)/2)} = 2 \frac{c^2 + \epsilon c}{(c + \epsilon)(c + c)} \approx 1. \quad (27)$$

Likewise,

$$s(y, z) = \frac{\phi_1(y)\phi_1(z)/2 + \phi_2(y)\phi_2(z)/2}{(\phi_1(y)/2 + \phi_2(y)/2)(\phi_1(z)/2 + \phi_2(z)/2)} = 2 \frac{c\epsilon + c^2}{(c + c)(\epsilon + c)} \approx 1. \quad (28)$$

However,

$$s(x, z) = \frac{\phi_1(x)\phi_1(z)/2 + \phi_2(x)\phi_2(z)/2}{(\phi_1(x)/2 + \phi_2(x)/2)(\phi_1(z)/2 + \phi_2(z)/2)} = 2 \frac{c\epsilon + \epsilon c}{(c + \epsilon)(\epsilon + c)} \approx 4 \frac{\epsilon}{c} \ll 1 \quad (29)$$


Figure 5

Generative distributions that challenge metric assumptions. **A.** Triangle inequality violation. **B.** Nearest neighbor statistics violation. The different ϕ_i correspond to available distributions and the regions highlight their non-negligible support over stimuli x, y, z, c_i .

in clear violation of the triangle inequality.

Nearest Neighbor Statistics Violations

Beyond metric axioms, spatial models of similarity impose strong constraints on the statistics of nearest neighbors (i.e., the number of points that a given point is closest to, and hence most similar). Such constraints, like those of metric axioms, may be incompatible with semantic organization, especially when relying on lower-dimensional representations (Tversky & Hutchinson, 1986). This problem is particularly pronounced in semantic domains that possess superordinate categories that are highly similar to many individual concepts. To make this concrete, consider the problem of embedding the concepts *fruit*, *apple*, *orange*, and *banana* in a one-dimensional space such that the most similar concepts fall closest to each other. Since *fruit* is the superordinate category, it should be the most similar (i.e., nearest neighbor) to all other concepts, but this is impossible in one dimension because a given point can be nearest neighbor to at most two other points (the two points that are directly to its left and right). Tversky and Hutchinson (1986) argued that this problem of “hubness” extends to higher-dimensional spatial

representations for a broad class of point distributions.

We will show next that generative similarity can capture this ‘hubness’ property.

Consider, for example, a set of stimuli $c_0, c_1, \dots, c_n$ (e.g. concepts) for some large n subject to the following generative process: we first uniformly sample a distribution from a set of n available distributions ϕ_i where $i = 1, \dots, n$ and then sample stimuli from that distribution. Let us further consider the case in which each distribution ϕ_i satisfies $\phi_i(c_i) = \phi_i(c_0) = a$ for some finite constant a and $\phi_i(c_j) = \epsilon \ll a$ for all $j \neq i, 0$ (Figure 5B). Note that from normalization $\sum_{j=0}^n \phi_i(c_j) = 1$ (for any i) we have $a = (1 - (n - 1)\epsilon)/2$ so that $a \gg \epsilon$ implies⁴ $n\epsilon \ll 1$ (i.e., it is infinitesimal) and therefore $n\epsilon \ll a$ (since a is finite by construction). Given this model, we can next compute generative similarity between the different stimuli using Equation (6). Indeed,

$$s(c_0, c_{i \neq 0}) = \frac{[(n - 1)\epsilon a + a^2] / n}{(na) / n \times [(n - 1)\epsilon + a] / n} \approx \frac{a^2}{a^2} = 1. \quad (30)$$

Likewise,

$$s(c_{i \neq 0}, c_{j \neq i, 0}) = \frac{[(n - 2)\epsilon^2 + 2\epsilon a] / n}{[(n - 1)\epsilon + a] / n \times [(n - 1)\epsilon + a] / n} \approx \frac{2n\epsilon}{a} \ll 1. \quad (31)$$

Thus, under this generative model we have $s(c_0, c_i) \gg s(c_j, c_i)$ for any $j \neq i$ which means that c_0 is the nearest neighbor to all other stimuli as desired.

By capturing violations of symmetry, the triangle inequality, constraints on nearest-neighbor statistics, as well as other foundational results such as Shepard’s law of generalization, generative similarity provides the right level of generality to unify a wide array of theoretical results from the literature on similarity. In all cases, this is achieved by specifying a transparent generative model of the data and then using Bayesian inference to reason about the distributions that generated various stimuli. These results suggest that generative similarity may provide an effective foundation for exploring the representations that allow people to make useful generalizations in different domains.

⁴ This follows from demanding $a = (1 - (n - 1)\epsilon)/2 \gg \epsilon$, rearranging the terms, and using the fact that n is large so that $n - 1 \approx n$.

Learning Representations that Support Generalization and Abstraction

Moving beyond theoretical considerations, we will next show how generative similarity can be used to learn representations that are empirically predictive of human behavior and that can support generalization and abstraction. Here we draw inspiration from Shepard’s work on multidimensional scaling and additive clustering (Shepard, 1980). We combine generative similarity with contrastive learning (Chen et al., 2020), a framework that derives representations by embedding similar stimuli close together and dissimilar stimuli far apart. The benefit here is threefold: (i) the encoder that is trained by optimizing this objective can embed new stimuli that were not part of the training set, and thus enable generalization, (ii) the framework can be combined with Monte-Carlo techniques to allow for estimating generative similarity even when its analytical form is intractable, and (iii) since contrastive learning is widely used in machine learning, combining it with generative similarity provides a prescription for learning human-aligned representations in neural networks. We will next lay out the mathematical formulation, and then apply it to three cognitively-motivated domains.

Generative Similarity and Contrastive Learning

Given a stimulus domain $X \sim p(x)$ and a pairwise notion of similarity between stimuli, contrastive learning derives representations by optimizing a parametric family of neural network encoders $\phi_\varphi(x)$ to minimize a suitable contrastive loss function (Chen et al., 2020; Oord et al., 2018). The loss function measures the consistency between embedding similarity, as specified by some function (e.g., $s_{\text{emb}} = e^{-d}$ where d is distance), and the true similarity. To apply contrastive learning to generative similarity s_{gen} , we distinguish between two scenarios. When s_{gen} is analytically tractable (or if some distributional similarity proxy is available), we can directly optimize the encoder parameters such that the difference between generative similarity and the corresponding embedding similarity is

488 minimized, e.g.,

$$\varphi^* = \arg \min_{\varphi} \mathbb{E}_{(X_1, X_2)} (s_{\text{emb}}(\phi_{\varphi}(X_1), \phi_{\varphi}(X_2)) - s_{\text{gen}}(X_1, X_2))^2. \quad (32)$$

489 If, on the other hand, s_{gen} is not tractable, we can implicitly incorporate it in training by
 490 applying Monte Carlo sampling from the generative model to produce “same” and
 491 “different” pairs

$$(X, X^+, X^-) \sim \begin{cases} \theta^+, \theta^- \sim p(\theta), & \text{Sample distributions} \\ x, x^+ \sim p(x|\theta^+), & \text{Sample “same” examples} \\ x^- \sim p(x|\theta^-), & \text{Sample “different” examples} \end{cases} \quad (33)$$

492 which, in turn, can be incorporated into a triplet loss function $\ell(x, x^+, x^-)$ that seeks to
 493 maximize the embedding similarity between X, X^+ and to minimize it between X, X^-

$$\varphi^* = \arg \min_{\varphi} \mathbb{E}_{(X, X^+, X^-)} \ell(\phi_{\varphi}(X), \phi_{\varphi}(X^+), \phi_{\varphi}(X^-)) \quad (34)$$

494 For example, ℓ could be the difference between the encoding distance of the pairs (x, x^+)
 495 and (x, x^-) , i.e., $\ell(x, x^+, x^-) = d(\phi_{\varphi}(x), \phi_{\varphi}(x^+)) - d(\phi_{\varphi}(x), \phi_{\varphi}(x^-))$ for some distance
 496 measure of choice $d(\cdot, \cdot)$. Additional mathematical details are provided in Appendices E
 497 and F, followed by example simulations in an analytically tractable case in Appendix G.

498 Our goal in what follows is to apply this formalism to domains of increasing
 499 complexity to evaluate the impact of generative similarity on capturing complex human
 500 behaviors. In particular, we consider: (i) quadrilateral shapes (Sablé-Meyer et al., 2021),
 501 (ii) abstract relational geometric concepts parameterized by probabilistic programs (Hsu
 502 et al., 2022), and (iii) natural images sampled from semantic hierarchies (Deng et al., 2009).

503 As a baseline model, we use the popular contrastive learning framework SimCLR
 504 (Chen et al., 2020). SimCLR is a training framework whereby a representation is learned
 505 by applying various augmentations (i.e., transformations) to the training data to artificially
 506 construct positive (“same”) and negative (“different”) pairs that are then incorporated in a
 507 contrastive loss such as the one shown in Equation (34). This objective is equivalent to a

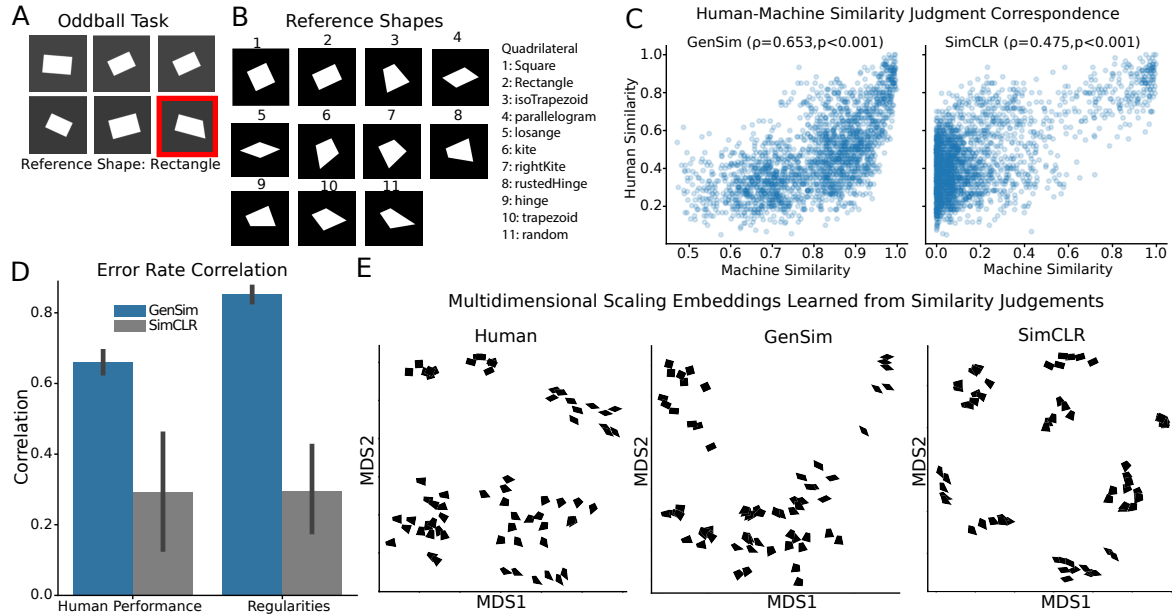
default generative process that is devoid of domain-specific knowledge whereby a base stimulus (e.g., an image) is sampled and then a random augmentation is applied. In particular, the SimCLR objective applies general augmentations to visual stimuli such as random cropping, rotations, and blurring, with the goal of making representations of an image and its augmented version as similar as possible, and representations of different images as dissimilar as possible. By using SimCLR as a baseline, we show the value of using contrastive learning with generative similarity in providing richer inductive bias to capture human behavior.

Application 1: Learning Abstract Geometric Regularity

Background

There is strong evidence that humans are uniquely sensitive to abstract geometric regularity (Henshilwood et al., 2011; Saito et al., 2014). Sablé-Meyer et al. (2021) compared diverse human groups (varying in education, cultural background, and age) to non-human primates on a simple oddball discrimination task. Participants were shown a set of five reference shapes and one “oddball” shape and were prompted to identify the oddball (Figure 6A). The reference shapes were generated using basic geometric regularities: parallel lines, equal sides, equal angles, and right angles, which can be specified by a binary vector corresponding to the presence or absence of these specific geometric features. There were 11 types of quadrilateral reference shapes with varying geometric regularity, from squares (most regular) to random quadrilaterals containing no parallel lines, right angles, or equal angles/sides (least regular; Figure 6B). In each trial, five different versions of the same reference shape (e.g., a square) were shown in different sizes and orientations. The oddball shape was a modified version of the reference shape, in which the lower right vertex was moved such that it violated the regularity of the original reference shape (e.g., moving the lower right vertex of a trapezoid such that it no longer has parallel sides). Figure 6A shows an example trial.

Sablé-Meyer et al. (2021) found that humans were naturally sensitive to these

**Figure 6**

*Generative similarity to instill human shape regularity biases. **A.** The oddball task of Sablé-Meyer et al. (2021) used six quadrilateral stimulus images, in which five images were of the same reference shape (differing in scale and rotation) and one was an oddball (highlighted in red) that diverged from the reference shape’s geometric properties. In this example, the reference shape is a rectangle; note that the oddball does not have four right angles like the rectangles. **B.** Different reference shapes used for the oddball task, ordered from most regular to least regular. **C.** Human vs. model similarity for GenSim and the SimCLR baseline. **D.** Correlation of error rates on the Oddball task across quadrilaterals with those of humans (right) and the number of geometric regularities of the reference shape (left). Error bars indicate standard errors. **E.** MDS embeddings derived from similarity in humans and the GenSim and SimCLR models.*

geometric regularities (right angles, parallelism, symmetry, etc.) whereas non-human primates were not. Specifically, they found that human performance was best on the oddball task for the most regular shapes, and systematically decreased as shapes became more irregular. Conversely, non-human primates performed well above chance, but they

performed worse than humans overall and, critically, exhibited no influence of geometric regularity. Additionally, Sablé-Meyer et al. (2021) tested pretrained convolutional neural network (CNN) models such as CorNet (Kubilius et al., 2019) on the task. CorNet (Core Object Recognition Network) is a convolutional neural network model with an architecture that explicitly models the primate ventral visual stream. It is pretrained on a standard supervised object recognition objective on ImageNet and is one of the top-scoring models of “brain-score”, a benchmark designed to test models of the visual system using both behavioral and neural data (Schrimpf et al., 2018). Like the monkeys, CorNet exhibited no systematic relationship with the level of geometric regularity. We aim to show the utility of generative similarity in instilling abstract geometric regularity biases into such neural networks and to better predict human similarity judgments over the quadrilateral domain.

Methods

Generative Similarity. The shape categories of Sablé-Meyer et al. (2021) can be specified by binary feature vectors corresponding to the presence or absence of abstract geometric features (equal angles, equal sides, parallel lines, and right angles of the quadrilateral) from which individual examples can be sampled. Formally, we can use these features in the Beta-Bernoulli generative process we outlined in Figure 4A and take the limit in which these features are fixed within a given quadrilateral class (since these geometric classes are defined by the truth value of their features). In Appendix H, we show that in this limit the generative similarity measure between shapes σ_1, σ_2 with feature vectors $f_i^{(1)}, f_i^{(2)}$ is given by $\log s(\sigma_1(f^{(1)}), \sigma_2(f^{(2)})) \propto -\sum_i (f_i^{(1)} - f_i^{(2)})^2$.

Model Training and Evaluation. We used the aforementioned generative similarity measure to instill the geometric regularity effect into CorNet, the same model Sablé-Meyer et al. (2021) used in their experiments to show conventional CNNs fail to reproduce the regularity effect. Given a random pair of quadrilateral stimuli from Sablé-Meyer et al. (2021), we computed the above quantity using their respective binary geometric feature vectors and finetuned the pretrained CorNet model on a contrastive learning objective

using these distances. Specifically, we first calculated the Euclidean distance of the model’s final layer embedding between different quadrilateral images, then calculated the Euclidean distance between the quadrilaterals’ respective geometry feature vectors, and finally used the mean squared error between the embeddings’ distance and the feature vectors’ distance as the loss. The geometric feature vectors were a set of 22 binary features encoding the following properties: 6 features per pair of edges encoding whether their lengths are equal or not, 6 features per pair of angles coding whether their angles are equal or not, 6 features per pair of edges encoding whether they were parallel or not, and 4 features per angle encoding whether they were right angles or not. This pushed quadrilaterals with similar geometric features together and pulled those with different geometric features apart in the model’s representation (see Appendix I for additional technical details, including the training of the SimCLR baseline). Like Sablé-Meyer et al. (2021), to test the model on the oddball task, we extract the embeddings for all 6 choice images and choose the oddball as the one that is furthest (i.e. Euclidean distance) from the mean embedding. In addition to the oddball task, we evaluated the extent to which the trained model can predict direct pairwise similarity judgments by computing cosine similarity between the embeddings at the final layer ($\cos(u, v) = u \cdot v / \|u\|_2 \|v\|_2$ which varies between -1 and 1) and correlating the result against human judgments.

Participants. For the oddball task studied by Sablé-Meyer et al. (2021), we used the data from Experiment 1. This comprised online data from $N = 612$ French participants (395 male, 217 female; reported age statistics: <18 years, 42 participants; 18 - 25 years, 127 participants; 25 - 60 years, 419 participants; >60 years, 24 participants). In addition to the human data, we used the data collected from $N = 26$ Guinea baboons (18 female, 8 male; reported age statistics: *Mean* = 11, *Range* = 1.5 – 23 years; Sablé-Meyer et al., 2021). For the similarity task, we recruited $N = 214$ US participants from Prolific⁵ (97 self-identified as man, 103 as woman, and 14 as other; age statistics: *Mean* = 38.7, *SD* = 12.1,

⁵ <https://www.prolific.com/>

Range = 18 – 78). Participants were paid 12 USD per hour and provided informed consent prior to participation in accordance with an approved Princeton University Institutional Review Board Protocol (#10859).

Stimuli. The quadrilateral shapes comprised 11 classes that varied in their degree of regularity and were labeled as: square, rectangle, isoTrapezoid, parallelogram, losange, kite, rightKite, rustedHinge, hinge, trapezoid, and random (Figure 6B; see Supplementary Table I1 for list of regularities). In the oddball task, 5 variants of the same reference shape (class) were produced by applying rescaling and rotations, and an additional oddball shape was produced by moving the lower right vertex of the reference shape in order to violate its regularity (Figure 6A; additional details can be found in Sablé-Meyer et al., 2021). For the similarity task stimuli, we randomly selected 6 shapes from each category so that overall we had 66 unique shapes and correspondingly $66 \times 65/2 = 2145$ unique pairs.

Procedure. In the oddball task, participants completed two practice trials followed by 44 test trials. In each trial, participants were instructed to click on the outlier as quickly as possible, after which they received visual feedback on whether their answer was correct or not (by highlighting the correct shape in green and the incorrect shape in red; the study and instructions can be accessed via the following link provided by the original authors⁶; additional details regarding data collection with baboons can be found in the Supplementary Information of Sablé-Meyer et al., 2021). For the similarity task, participants received the following instructions: “In this study, we are studying how people perceive geometric shapes. Your task is to rate the similarity between different pairs of geometric shapes. You will have seven response options, ranging from 0 (‘Completely Dissimilar’) to 6 (‘Completely Similar’). Choose the one you think is most appropriate. Note: no prior expertise is required to complete this task, just choose what you intuitively think is the right answer.” Participants then saw 9 random shapes to help familiarize them with the stimuli. They then completed up to 100 trials where they responded to the query

⁶ https://neurospin-data.cea.fr/exp/mathias-sable-meyer/oddball_original/

“How similar are the following two white geometric shapes?”. The pairs were randomly selected, and the order of the images within each pair was randomized. Overall, we collected 20,948 similarity judgments with an average of 10 judgments per pair. We averaged the responses per pair and divided by a factor of 6 (so that values vary between 0 and 1) to derive the final similarity matrix (i.e., each element s_{ij} encodes the similarity between stimulus i and j). We confirmed in a post-hoc analysis that the data was sufficiently powered by computing the split-half inter-rater reliability (IRR) bootstrapped over trials with 1,000 repetitions and applying a Spearman-Brown correction $2r/(1+r)$ where r is the Pearson correlation between the data splits (Brown, 1910). We found that the IRR was .84 (95% CI [.83, .85]).

MDS Visualization. Here and in later sections, we provide two-dimensional multidimensional scaling (MDS) visualizations of the human and model similarity matrices. MDS solutions were derived using the `scikit-learn` MDS method (Pedregosa et al., 2011) with a tolerance parameter of $1e-100$, and maximum iteration threshold of 10,000. We first applied metric MDS to derive an initial configuration which we then fed into a non-metric MDS run to derive the final embeddings.

Transparency and Openness. This work was conducted in accordance with an approved Institutional Review Board protocol (Princeton University, #10859). All data and code can be accessed via the following OSF link.⁷ The study was not preregistered.

Results

We first looked at how each model training approach can approximate human similarity (Figure 6C). Specifically, we collected human similarity judgments of the quadrilateral stimuli from Sablé-Meyer et al. (2021) (see Methods) and measured their correspondence with similarity scores derived from a CorNet model trained with generative similarity (GenSim) and simple baseline contrastive learning (SimCLR). To quantify similarity for the models, we used the cosine similarity between image embeddings, a

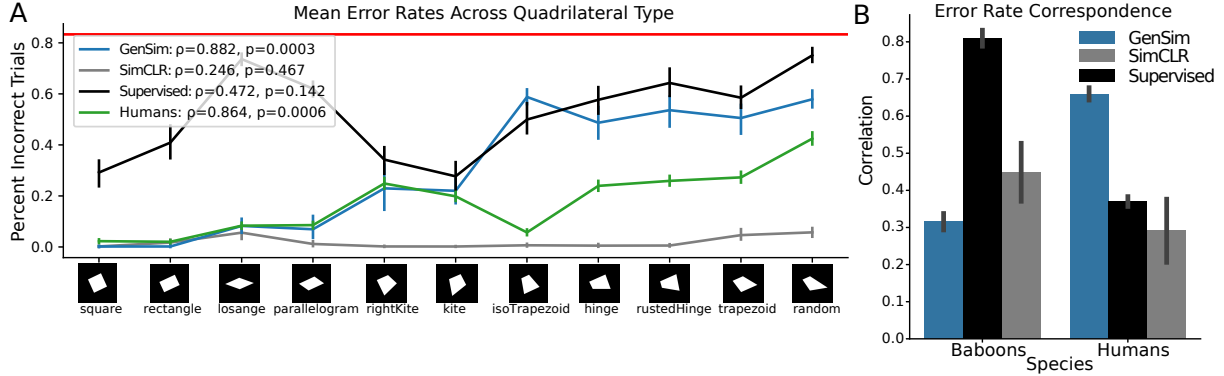
⁷ https://osf.io/srmn3/overview?view_only=1d55cecc30ef4b9b9156267480c83c7d

standard way to measure similarity in high-dimensional embedding spaces learned in machine learning models (Mikolov et al., 2013). We found that this correspondence was higher for the GenSim model (Spearman’s $\rho = 0.653$, $p < 0.001$) than the baseline SimCLR model (Spearman’s $\rho = 0.475$, $p < 0.001$).

The geometric regularity effect observed for humans in Sablé-Meyer et al. (2021) was a close correspondence between the number of geometric regularities and error rates. For example, humans performed best on the most regular shapes, such as squares and rectangles and worst on the least regular shapes, such as trapezoids. In Figure 6D (right), we show the correspondence between these regularities and the error rates of a CorNet model finetuned on generative similarity (GenSim) and baseline contrastive learning (SimCLR). We also show the error rate correspondence between the two models and adult human participants (Figure 6D, left) from the experiment of Sablé-Meyer et al. (2021). Like humans, the model error rates for the GenSim CorNet model significantly increase as the shapes become more irregular (Spearman’s $\rho(9) = 0.88$, $p = 0.0003$). This is not the case for the model finetuned with SimCLR (Spearman’s $\rho = 0.246$, $p = 0.467$).

To get a visual sense of this correspondence, we apply multidimensional scaling (MDS; Shepard, 1980) to the resulting similarity matrices of humans and trained models (Figure 6E). SimCLR seems to put each quadrilateral category into completely independent and separate cluster. Although humans do this as well, these clusters are more “blended together” for irregular stimuli (e.g. trapezoids, hinges, etc.). The GenSim model also exhibits a similar pattern, better reflecting the human psychological space than the baseline SimCLR model, consistent with the quantitative analysis.

Sablé-Meyer et al. (2021) in their original work compared human and monkey error rates in the oddball task to CorNet models that were finetuned on a supervised classification objective. We show this comparison in Figure 7A, along with those of a CorNet model trained with GenSim and SimCLR finetuning objectives (see Appendix I for more details on the training of our models). Again, like humans, the model error rates for


Figure 7

Error rates for individual quadrilateral types across models. **A.** Mean error rates per quadrilateral type on the oddball task, ordered from most regular to least regular shapes. Sablé-Meyer et al. (2021) found that human error rates systematically increase as the regularity of the reference quadrilaterals decreases, whereas those of non-human primates do not. We use this same evaluation for a CNN model with different finetuning objectives, reporting the Spearman rank correlation between model performance and number of geometric regularities across quadrilateral type (see Supplementary Table I1 for these values). Only the model with the generative similarity (GenSim) finetuning objective exhibited a significant human geometric regularity effect. Error bars denote standard errors over 10 training runs, red line shows chance-level error rate. **B.** Correlation between model error rates and human or monkey error rates. Error bars are standard errors. Supervised is the most monkey-like whereas GenSim is the most human-like.

671 the GenSim CorNet model significantly increase as the shapes become more irregular
 672 (Spearman’s $\rho(9) = 0.88$, $p = 0.0003$). This is not the case for the model finetuned with
 673 supervised classification (Spearman’s $\rho(9) = 0.472$, $p = 0.142$), highlighting the fact that
 674 not all supervision can induce the right human inductive bias. When correlating the error
 675 rates of the finetuned models with those of humans and monkeys (Figure 7B), we see an
 676 interesting double dissociation between the GenSim and supervised settings. Specifically,
 677 the GenSim-trained CorNet model’s error rates match human error rates significantly more

than monkey error rates ($t(18) = 12.45, p < 0.0001$), whereas, in line with the findings of Sablé-Meyer et al. (2021), those of the supervised CorNet model match monkey error rates significantly more than human error rates ($t(18) = 17.43, p < 0.0001$).

Application 2: Learning Abstract Geometric Concepts and Relations Defined by Probabilistic Programs

Background

Our next domain builds on recent work that shows the limitations of current vision models in understanding abstract concepts of geometric stimuli, a capability that humans possess (Hsu et al., 2022). Hsu et al. (2022) proposed a benchmark based on a probabilistic programming language that synthesizes stimuli that capture intuitive geometric concepts. The Geoclidean Domain-Specific Language (DSL) is directly inspired by the work of Euclid, who formalized geometric axioms using a compass and a straight edge. The DSL contains three simple primitives reflecting the principles of Euclidean geometry – a point with an (x, y) location, a line, which is parameterized by two points, and a circle, which is parameterized by a center point and an edge point. Lines and circles are parametrized by points while points can be bound to previously built objects. This DSL can define a distribution of probabilistic programs that can then be rendered into images (see Figure 8A). Each program can be thought of as the specification for a set of geometric objects that have complex relations among them.

Hsu et al. (2022) defined a set of 37 programs within this DSL that resemble concepts defined in Euclid’s *Elements* (Simson, 1838; programs provided in Appendix J). They then designed a few-shot generalization task where participants or models are given examples from a reference program and have to use these examples to correctly classify other images from the reference programs (positive examples) and images from different programs (negative examples). There were two types of programs that rendered the negative examples (Figure 8A): “close”, which had minimal changes with respect to the reference program, and “far”, which had a greater number of changes to the reference

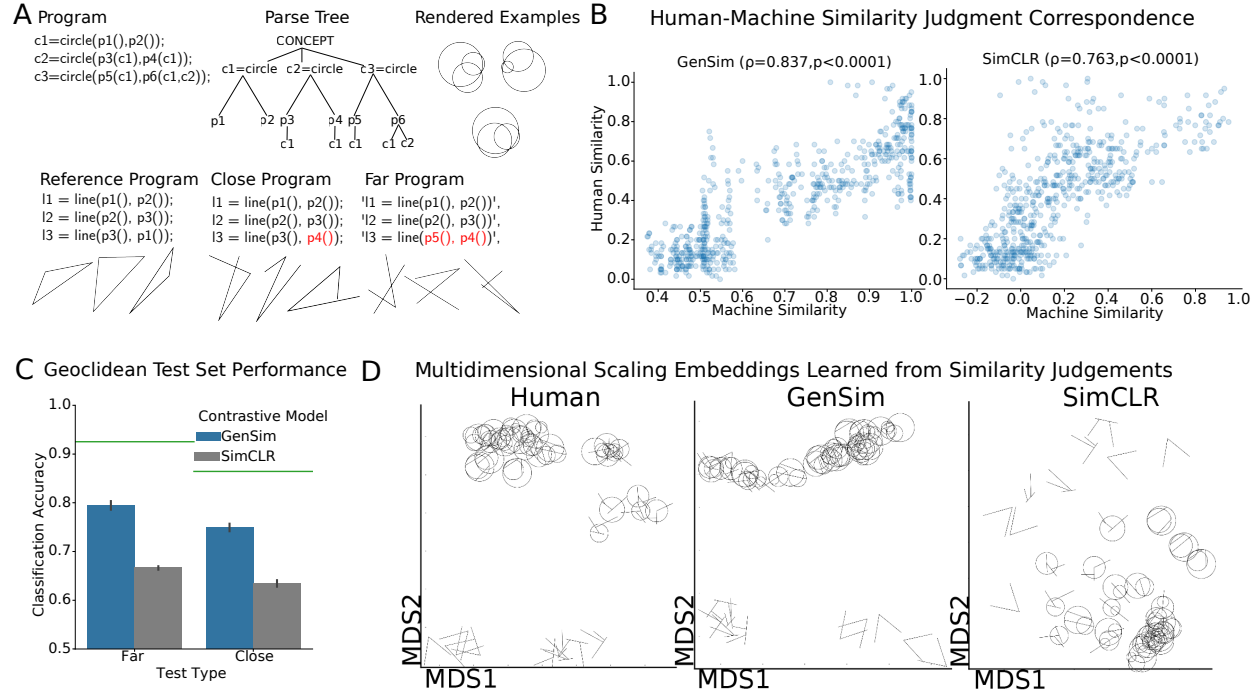


Figure 8

Generative similarity over probabilistic programs enables few-shot reasoning of abstract geometric concepts. **A.** Example program, program parse tree, and rendered generations from the Geoclidian DSL. In the Geoclidian classification task, participants or models are given examples from a reference program. They have to successfully distinguish between examples from the reference program and two types of negative examples – ones from a “close” program and ones from a “far” program (see Methods). **B.** Correspondence between human similarity judgments and model similarity scores (as determined by cosine similarity between image embeddings). **C.** Performance of contrastive embeddings in correctly classifying reference program images. Green lines indicate mean human accuracy and error bars denote standard error over training runs. **D.** MDS embeddings derived from similarity judgments in humans and both GenSim and the SimCLR baseline.

705 program. This task is designed to probe the intuitive understanding of geometry, as
 706 evidenced by the ability to generalize from a limited set of examples to new instances. Hsu
 707 et al. (2022) demonstrated that humans show a robust capacity for such generalization

across 37 different concepts, suggesting a natural sensitivity to the hierarchical structure and part relations inherent in Euclidean geometry. Conversely, they found that pretrained vision models struggle with few-shot generalization in this context. We will show next how generative similarity can close the gap by inducing more human-like representations.

Methods

Generative Similarity. Computing generative similarity over a process in which probabilistic programs are sampled and then stimuli are generated from those programs is challenging and does not admit a closed-form formula. However, we can still derive representations by relying on tractable proxies and Monte Carlo sampling. Indeed, the syntactic structure of the Geoclidian programs sufficiently describes the relations between the primitive objects (point, circle, and line) necessary to correctly categorize images as belonging to a particular program. Given this, we formulated a Context-Free Grammar (CFG) from the Geoclidian DSL that can parse the Geoclidian programs into parse trees (see Appendix K for the full CFG grammar). If we interpret the Geoclidian programs as samples from this CFG, their generative similarity will be based on sharing the same production rules or, equivalently, shared substructures in their parse trees. As an implementation of this measure, we used the Fast Tree Kernel (FTK) by Moschitti (2006), which recursively computes the number of shared subtrees in two parse trees to determine their similarity. Thus, generative similarity between two images becomes that of the parse tree similarity of their underlying programs. We note that an alternative approach based on Monte-Carlo sampling could instead randomly sample programs and then produce same-different image triplets from them for training (Equations (33) - (34)). This approach is closely related to supervised contrastive learning (SupCon; Khosla et al., 2020) and we explore it in Appendix O.

Model Training and Evaluation. The loss function used for model training was the mean squared error between the cosine similarity of two rendered images’ embeddings and the tree kernel similarity of the images’ respective generative programs. We used the 37

geometric concepts presented in the original Geoclidian work for our experiments (see Appendix J for the full list and their respective programs). We randomly held-out 7 concepts for testing our models and generated a dataset of 100,000 images that were random samples from the 30 training concepts (around 3000 images per concept). We used a pretrained ResNet-50 architecture for our experiments (He et al., 2015), a model shown in the original Geoclidian work to do poorly at the benchmark. As before, we had a baseline model trained on the SimCLR objective (Chen et al., 2020). The SimCLR objective produces augmented versions of an image (e.g. random cropping, rotations, Gaussian blurring, etc.) with the goal of making representations of an image and its augmented version as similar as possible and representations of different images as dissimilar as possible. See Appendix L for additional technical details. Beyond the task of Hsu et al. (2022), we also evaluated how well each model is capable of predicting direct human similarity judgments by correlating the cosine similarity of embeddings against human ratings.

Participants. For the few-shot generalization task of Hsu et al. (2022), the authors recruited $N = 30$ Prolific participants. Participants were required to be at least 18 years of age, however, no further demographic information was provided. For the similarity judgment task, we recruited $N = 65$ US participants from Prolific (33 self-identified as man, 29 as woman, and 3 as other). The age statistics were: $Mean = 40.0$, $SD = 12.8$, $Range = 20 - 73$ years. Participants were paid at a rate of 12 USD per hour, and they provided informed consent prior to participation in accordance with an approved Princeton University IRB protocol (#10859).

Stimuli. The stimuli for the few-shot generalization task comprised 5 reference shapes per concept that were randomly generated from each program provided in Appendix J (example stimuli are shown in Figure 8). In addition, there were 15 test stimuli per concept that were divided into 5 positive examples (i.e., generated from the same concept program), 5 “close” negative examples (generated by altering a small number of the constraints of the

reference program), and 5 “far” negative examples (generated by altering a larger number of the constraints of the reference program; additional details can be found in Hsu et al., 2022). As for the similarity task, it comprised 35 unique stimuli (i.e., $35 \times 34/2 = 595$ unique pairs) corresponding to 5 realizations from each of the 7 held-out concepts that were not part of the model training sets (indicated with four asterisks in Appendix J).

Procedure. In the few-shot generalization task (Hsu et al., 2022), participants covered all concept sets in random order. In each concept-set trials corresponding to the 15 test stimuli, participants received the following the instructions: “Each of the five images shown is a ‘wug’. Is the image below a ‘wug’ or not a ‘wug’?”, where ‘wug’ corresponded to a random nonsense word (per concept set). Participants provided their response by selecting an option. As for the similarity task, participants received the following instructions: “In this study, we are studying how people perceive geometric relations. Your task is to rate the similarity between different pairs of geometric relations. You will have seven response options, ranging from 0 (‘Completely Dissimilar’) to 6 (‘Completely Similar’). Choose the one you think is most appropriate. Note: no prior expertise is required to complete this task, just choose what you intuitively think is the right answer.” Participants were then present with 9 random stimuli to help familiarize them with the shapes. Then, they completed up to 100 trials, each responding to the query: “How similar are the following pair of geometric relations?”. The pairs were randomly selected, and the order of the images within each pair was randomized. In total, we collected 5,950 similarity judgments with an average of 10 judgments per pair. As before, we averaged the responses per pair and divided by a factor of 6 to derive the final similarity matrix. The resulting split-half IRR was .93 (95% CI [.92, .94]).

Results

We begin by analyzing the models’ ability to predict human similarity judgments over held-out geometric concepts (see Methods). The results are shown in Figure 8B. We found that both the generative-similarity-trained model (GenSim) and the SimCLR-trained

baseline provided high correlations with human data ($\rho = 0.837$, $p < 0.0001$ and $\rho = 0.763$, $p < 0.0001$, respectively) with GenSim providing greater correspondence (see Appendix L for more details on the training of our models). This can also be seen visually in the MDS plots of the similarity data (Figure 8D). In particular, while the human and GenSim MDS solutions appear to separate certain abstract classes more strictly (e.g., different arrangements of lines, or relations between lines and circles), SimCLR appears to smoothly organize the stimuli based on their low-level properties.

Moving next to the few-shot generalization task of Hsu et al. (2022) which requires a finer sensitivity to geometric relations, we used test programs held out from training to evaluate our models using the exact same procedure as Hsu et al. (2022) (Figure 8C; see Methods). In particular, we computed the mean embedding across 5 reference images from a test program (“the prototype”) and measured the distance between a given image’s embedding (to be classified; see Procedure) and the program prototype. The image was then classified by comparing its distance value against a certain threshold that was fit to maximize model classification accuracy (Hsu et al., 2022, did this to bias accuracy in favor of the models over humans, and we replicate this process across all of our models to ensure fair comparison across models).

The GenSim model, though still below human accuracy, substantially outperformed SimCLR at classification (GenSim: $79.5\% \pm 0.9$ Far image accuracy and $74.9\% \pm 0.78$ Close image accuracy; SimCLR: $66.6\% \pm 0.4$ Far image accuracy and $63.4\% \pm 0.67$ Close image accuracy; errors are standard errors). To our knowledge, this is the current state-of-the-art accuracy on the Geoclidian benchmark, although it is important to note that Hsu et al. (2022), did not finetune their models on the task. In the supplement (Figure O1), we also compare our models against the Monte-Carlo approximation approach (see Methods) which closely aligns with supervised contrastive learning (Khosla et al., 2020). We found that this model achieves comparable performance on the similarity task, and falls between SimCLR and GenSim in the few-shot generalization task, consistent with our expectation.

Application 3: Learning Semantic Hierarchies Underlying Natural Images

Background

Humans leverage complex semantic hierarchies to reason about their sensory environment (e.g., living organism $\rightarrow$ animal $\rightarrow$ mammal $\rightarrow$ dog; Murphy, 2004; Peterson et al., 2018; Tenenbaum et al., 2011). Consequently, these hierarchical structures are reflected in many large machine learning datasets, most notably ImageNet (Deng et al., 2009) which builds on the WordNet⁸ object hierarchies originating in cognitive science (Fellbaum, 1998; Miller, 1995; Figure 9A). However, most vision models trained on this dataset do not take into account relations between categories in the hierarchy but rather treat categories independently. We will next show how training models on generative similarity with contrastive learning can enable capturing such semantic hierarchies and better predicting human judgments.

Methods

Generative Similarity. Since ImageNet stimuli are organized into a tree hierarchy according to WordNet (Fellbaum, 1998; Miller, 1995), a natural generative process for sampling images involves probabilistically traversing the tree hierarchy from the root to a leaf (i.e., a class without any additional subcategories; Figure 9A), and then uniformly sampling an image that is associated with that leaf (and its parent categories). At each level k in the tree hierarchy, we uniformly sample a branch with probability p_k (i.e., one over the number of available branches at that level) to proceed on to the next level. A single image can thus be described by multiple categories associated with different tree levels (e.g., a spider image is also an arthropod as well as an invertebrate and an animal). In Appendix M, we show that the generative similarity between two images x_i and x_j under the aforementioned sampling process is given by

$$s(x_i, x_j) = \frac{1}{K} \left(\frac{1}{p_1} + \frac{1}{p_1 p_2} + \cdots + \frac{1}{p_1 p_2 \cdots p_K} \right) \quad (35)$$

⁸ <https://wordnet.princeton.edu/>

where K is the depth of the lowest common category (or “ancestor”) between x_i and x_j (i.e., their “separation depth”) and $p_1, \dots, p_K$ are the probabilities of choosing each branch along the path (which we take to be uniform per branching point). Note that since $p_i \leq 1$ it follows that the deeper the separation depth between two stimuli, the larger their similarity will tend to be (for example, in the case of a binary tree where $p_i = 1/2$ for all i we have $s(x_i, x_j) = 2(2^K - 1)/K$ which is exponentially increasing).

Model Training and Evaluation. We defined a loss function based on the mean squared error between the embedding cosine similarity of image pairs x_i, x_j and $\log[s(x_i, x_j)]$ where $s(x_i, x_j)$ is their generative similarity. To show that generative similarity with contrastive learning can enable learning of embeddings that encode hierarchal image categories, we trained a ResNet-50 model on the standard ImageNet-1k⁹ dataset using this loss function. Moreover, we used SimCLR (with the same architecture and training data) as a domain-agnostic baseline. In Appendix O (see also Figure O2) we also consider a Monte-Carlo training scheme where we employ a triplet loss based on image categories (i.e., WordNet leaves; one anchor image, one positive image from the same category, and one negative image from a different category) which is equivalent to supervised contrastive learning (Khosla et al., 2020). See Appendix N for more details on the training of our models.

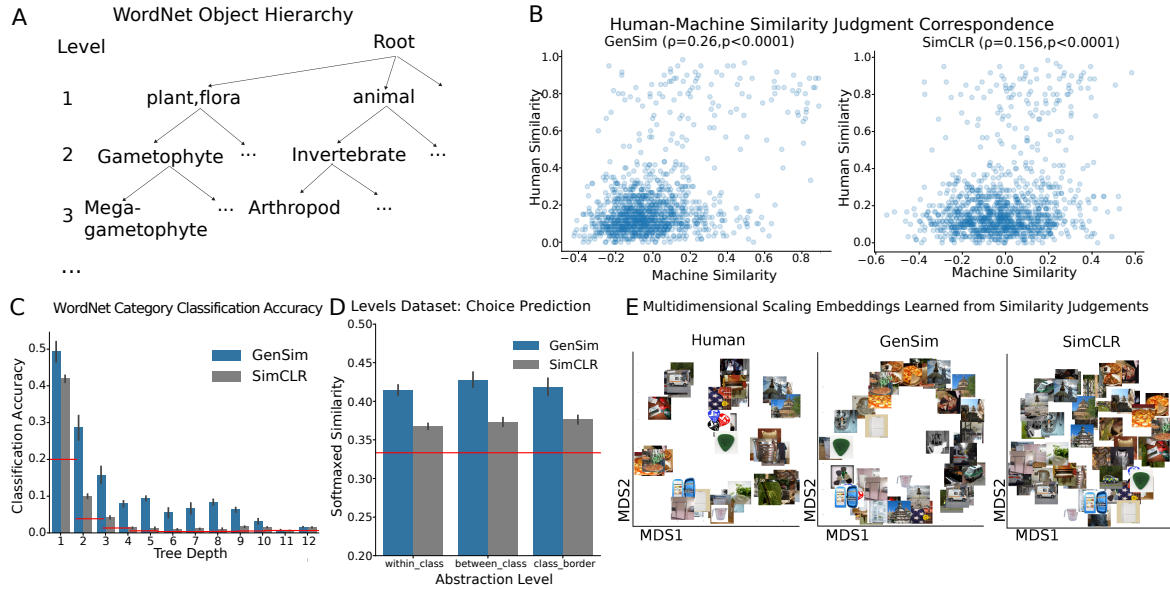
To evaluate the ability of the learned embeddings to encode information about hierarchical categories, we fit a Linear SVM (support vector machine; Cortes and Vapnik, 1995) on test set images to predict image categories for each tree level (i.e. we fit a different SVM for each level of the tree and each model). The SVM was fit for up to 100 epochs (stopping when the loss does not decrease by more than $1e-4$ for 5 epochs). We trained the SVM within a three-fold cross-validation loop, evaluating classification accuracy on a held-out test set, and then reported the mean across the three folds. To produce an aggregate score across levels that takes into account the fact that chance level is different

⁹ <https://huggingface.co/datasets/mlx-vision/imagenet-1k>

for each level depth (since there are more categories at deeper levels), we aggregated performance across tree levels via the metric $\sum_{i=1}^n \log \frac{a_i}{c_i}$, where a_i is the accuracy at tree level i , c_i is chance accuracy at tree level i , and n is the number of tree levels.

As for model evaluation against human data, we considered two datasets involving human judgments on ImageNet stimuli. First, the Levels Dataset of Muttenthaler et al. (2025) which comprises odd-one-out judgments on image triplets that span three different levels of abstraction: “coarse-grained” where each image is from a different ImageNet category, “fine-grained” where all images are from the same category, and “class-boundary” where two images are from the same category and one is from a different category. For each trial, if the participant chooses an option a out of three images a, b, c , we use our models’ embeddings to compute $\text{softmax}[\cos(a, b), \cos(a, c), \cos(b, c)]$ where $\cos$ is the cosine similarity between the respective images’ model embeddings. We then report the softmaxed $\cos(b, c)$ as the model’s likelihood for the participant’s choice (since the participant determined a is the odd-one-out). The softmax temperature was chosen as $\tau = \alpha \times rt$, where rt is the amount of time the human participant used to make a decision and α is a subject-specific parameter we grid-search over for each human participant to maximize the mean likelihood over choices across trials. We fit a separate α for each participant and each model to ensure a fair comparison across models. Second, we collected a dataset of pairwise similarity judgments over images from the ImageNet-1k test set (see below). We then correlated model embedding similarity against human similarity judgments as in prior experiments.

Participants. The Levels dataset (Muttenthaler et al., 2025) comprised data from $N = 473$ Prolific participants (202 female, 269 male, 2 other). The reported age statistics were $Mean = 31.8$, $SD = 8.0$ years. For the pairwise similarity task we recruited $N = 127$ US participants from Prolific (52 men, 66 women, 9 other; age statistics: $Mean = 41.3$, $SD = 13.5$, $Range = 21 - 77$ years). All participants provided informed consent prior to participation in accordance with an approved Princeton University IRB protocol (#10859),


Figure 9

*Generative similarity helps encode natural image hierarchies. **A.** WordNet (Fellbaum, 1998; Miller, 1995) gives a hierarchy that spans multiple semantic levels. The object categories in the classic ImageNet dataset (Deng et al., 2009) are sampled from different places on the WordNet tree. **B.** Correspondence between human similarity judgments and GenSim and SimCLR. **C.** Mean classification accuracy over chance across tree levels. Chance accuracies are denoted by red lines; note that these decrease as one goes deeper into the WordNet tree since the number of categories increases. Error bars denote standard error across model training runs. **D.** Mean model likelihood across human participants' choices on the Levels dataset (Muttenthaler et al., 2025), where participants did odd-one-out judgments at three different abstraction levels in which they were given three images and chose what they believe is the odd-one-out. The likelihood is based on computing the model embedding similarity between the pair of images the participant believed were not the odd-one-out (see Methods). We found that the GenSim model better accounted for human behavior. Error bars are standard errors across human subjects. **E.** MDS embeddings derived from human and model similarity.*

and they were paid at a rate of 12 USD per hour.

Stimuli. The ImageNet-1k dataset (Deng et al., 2009; Russakovsky et al., 2015) is one of the most commonly used subsets of ImageNet. It contains 1,000 object classes, 1,281,167 training images, 50,000 validation images, and 100,000 test images (additional details can be found here¹⁰). As for the Levels dataset (Muttenthaler et al., 2025), triplets were constructed by first grouping ImageNet into 717 course-grained WordNet superclasses to stratify sampling (e.g., all dog breeds would fall under the single category dog). Then, different triplet types were produced by sampling one, two, or three images from sampled classes (without replacement). In total, there were 450 trial sets, with each trial presented across 5 participants (additional details can be found in Muttenthaler et al., 2025). Finally, for the pairwise similarity stimuli, these comprised 50 images (i.e., $50 \times 49/2 = 1225$ unique pairs) which spanned 10 random categories with 5 random images per category from the ImageNet-1k test set.

Procedure. In the odd-one-out task of Muttenthaler et al. (2025), participants completed 330 experiment trials. In each trial, participants observed a fixation cross for 1 second, followed by a triplet. They were then instructed to select the “odd-one-out” image using the left, down, and right arrow keys on their keyboard. Responses were accepted within 1-15 seconds after triplet onset, and reaction time was recorded. The pairwise similarity task had an analogous structure to the previous domain. Upon providing informed consent, participants received the following instructions: “In this study, we are studying how people perceive images. Your task is to rate the similarity between different pairs of images. You will have seven response options, ranging from 0 (‘Completely Dissimilar’) to 6 (‘Completely Similar’). Choose the one you think is most appropriate. Note: no prior expertise is required to complete this task, just choose what you intuitively think is the right answer.” Participant then were presented with 9 random stimuli to help familiarize them with the domain. They then completed up to 100 random trials, in each responding

¹⁰ <https://huggingface.co/datasets/mlx-vision/imagenet-1k>

to the query “How similar are the following pair of images?”. Overall, we collected 12,250 human responses, with an average of 10 judgments per pair. The human IRR was .90 (95% CI [.89, .91]).

Results

The correspondence between the models’ pairwise embedding similarity and human similarity judgments is shown in Figure 9B (see Methods). As in the previous domains, we found that the generative similarity-trained model (GenSim) outperformed SimCLR in its ability to predict human data ($\rho = 0.260$, $p < 0.0001$, and $\rho = 0.156$, $p < 0.0001$, respectively; a visualization of the MDS solutions for the human and model similarity matrices are provided in Figure 9E). Although the correlations are moderate, we emphasize that the domain is high-dimensional and contains 1,000 categories, and our predictions were derived without training on any human data.

Next, to quantify how much the embeddings have encoded hierarchical categories, we used the learned models’ embeddings to train a Linear SVM classifier to predict categories from image embeddings at different tree depths (i.e. different levels of abstraction; see Methods). Individual accuracies across tree levels are shown in (Figure 9C). The model trained with generative similarity had overall higher accuracy in predicting categories across WordNet tree levels than the SimCLR baseline. To quantify that, we computed an aggregate score that normalizes accuracy relative to chance level at each depth (since there are more categories at lower levels; see Methods). As hoped, GenSim indeed produced a higher score ($score = 1.94$, $SE = 0.035$) than SimCLR ($score = 0.78$, $SE = 0.014$).

Finally, we evaluated the models’ ability to predict human behavior in odd-one-out judgments that span different levels of abstraction. To do so, we used the Levels dataset of Muttenthaler et al. (2025) (Figure 9D; see Methods). In this dataset, humans perform an odd-one-out judgment on image triplets that are sampled at three different levels of abstraction. In “fine-grained” triplets, all images are from the same ImageNet category. In

“coarse-grained”, they are from different categories. In “class-boundary”, two images are from the same category and one image is from a different category. We used a softmax function based on the embedding cosine similarity of each image pair to decide on the odd-one-out (see Methods). Our results, shown in Figure 9D, indicate that the GenSim model has a higher likelihood for human choices across all three triplet types. In particular, the mean ($\pm$ standard error) GenSim and SimCLR scores, respectively, were $0.415 \pm 0.006, 0.368 \pm 0.004$ for “fine-grained”, $0.428 \pm 0.01, 0.373 \pm 0.005$ for “coarse-grained”, and $0.419 \pm 0.011, 0.376 \pm 0.005$ for “class-boundary”.

In Supplementary Figure O2, we provide additional results for a supervised contrastive learning baseline (SupCon). Recall that this model corresponds to a generative process that relies on the lowest level of the WordNet hierarchy (i.e., sampling ImageNet classes and then sampling images from those classes to train on a triplet loss function). Interestingly, we found that the model resulted in higher correlations than GenSim in the pairwise similarity prediction task ($\rho = 0.392, p < 0.0001$), but underperformed GenSim in the category classification task across different tree depths ($score = 1.67, SE = 0.12$; GenSim performed significantly better than SupCon across different training runs, $t(8) = 2.225, p = 0.028$), as well as in predicting human behavior across participants on the odd-one-out task from the Levels dataset ($t(435) = 9.33, p < 0.0001$).

Discussion

We have provided a general Bayesian framework for formulating similarity as inference over generative processes. We showed that this approach can not only unify and illuminate classical results from the literature such as Shepard’s universal law (Shepard, 1987), Tversky’s feature model (Tversky, 1977), and Gaussian theories of categorization (Nosofsky, 1986) by tying each to specific choices of generative models, but, when combined with modern contrastive learning (Chen et al., 2020), it can also provide the right training signal for learning representations that better predict human judgments and support human-like generalization and abstraction.

Our generative framework provides a complementary account to the generalization-based formulation of similarity put forth by Shepard (1987) and its subsequent Bayesian reformulation by Tenenbaum and Griffiths (2001). Indeed, the generative account provides comparable explanatory power to that of Tenenbaum and Griffiths (2001), unifying featural and spatial models of similarity and allowing extensions to multi-item settings, but it also avoids the problem of relying on “blank” properties (e.g., having an enzyme; Kemp and Tenenbaum, 2009; Osherson et al., 1990) that are needed to specify similarity as generalization in a default context. Instead, similarity is directly (and transparently) linked to the agent’s assumptions about how stimuli are generated. Nonetheless, we should highlight that the two accounts are not necessarily at odds, especially if generative assumptions are tied to possessing certain properties (e.g., having whiskers and being generated from the distribution of cats).

Our work also extends the line of psychological research aimed at deriving representations from similarity data, as exemplified by techniques such as multi-dimensional scaling (Shepard, 1980) and additive clustering (Shepard & Arabie, 1979). By combining generative similarity with a suitable algorithmic framework, namely contrastive learning (Chen et al., 2020; Oord et al., 2018), we are able to harness the flexibility of modern machine learning for deriving representations. In particular, the generative-contrastive framework enables (i) deriving encoders that can embed new unseen data, (ii) handling cases in which not all pairwise stimulus similarities are available, (iii) capturing complex domains that are parametrized by probabilistic programs, and (iv) leveraging psychological insight to train human-aligned models.

Our findings support and significantly expand previous explorations of generative similarity (Destler et al., 2023; Kemp et al., 2005). We extend the work of Kemp et al. (2005) by systematically situating generative similarity within the broader literature, delineating its connection to other Bayesian accounts (Tenenbaum & Griffiths, 2001), as well as to foundational findings (e.g., Shepard’s universal law and metric violations in

semantic organization; Shepard, 1987; Tversky and Hutchinson, 1986). We also provide concrete strategies for handling cases in which the Bayesian computations are not tractable, a necessary ingredient for many practical applications of generative similarity (e.g., in the case of geometric relations that are parameterized by probabilistic programs discussed earlier). We also extend the work of Destler et al. (2023) who explored generative similarity in the context of skeleton-based accounts of shape similarity. By combining generative similarity with contrastive learning, we are able to significantly expand its applicability to other domains by harnessing modern machine learning.

Our work also has implications for the growing literature on representational alignment in machine learning (Sucholutsky et al., 2023). As noted earlier, contrastive learning is a widely used framework for representation learning (Chen et al., 2020; Gao et al., 2021; Khosla et al., 2020; Oord et al., 2018), and some recent work combines it with substantial amounts of human data (e.g., similarity judgments) to achieve better human alignment (e.g., Hebart et al., 2020; Jha et al., 2023; Muttenthaler et al., 2024). Our work suggests that when a psychologically-motivated generative model is available, one can leverage generative similarity and contrastive learning to improve human alignment without relying on any expensive and large-scale human data. Moreover, our experiments in the quadrilateral domain suggest that not all intuitive types of supervision necessarily lead to more human-aligned representations (since we found that training on generative similarity, but not on a supervised classification objective, lead to human-like error patterns; Figure 7).

Limitations and Future Directions

We end by discussing some potential limitations of the present work, how they may be addressed, and future research directions that they point towards.

Specifying a Hierarchical Generative Model

The starting point of our formulation is a hierarchical generative model of the data where distributions are sampled and then stimuli are generated from them. Specifying such

a model may not always be straightforward, and in some cases it may not be applicable (at least without some modifications). For example, consider domains such as monochromatic colors or tone pitch, both of which may be thought of as a single distribution capturing some natural statistics, with the similarity between pairs deriving from low-level perceptual processing constraints. Although this may indeed be the leading contributor to similarity in such domains, it is conceivable that generative considerations would still play a role in shaping the representation given a suitable context (e.g., by reasoning over color concepts in language; Winawer et al., 2007; Zaslavsky et al., 2018; or over musical notation systems and repertoires; Jacoby et al., 2019). Future research could look into disentangling such contributions to perceptual similarity and delineating the role of generative similarity specifically in the organization of low-level sensory domains. Another important class of extensions to consider is that of temporal processes (e.g., Markov processes). While these do not admit a simple hierarchical structure at face value, generative-contrastive techniques have recently been proposed in the machine learning literature to address such settings (intuitively, two states x_0 and x_T are similar to the extent that a temporal process starting at x_0 would also generate x_T ; Eysenbach et al., 2024). Connecting such contrastive learning techniques to cognitive modeling is a promising avenue for future work.

Uncertainty in the choice of generative model could be addressed via additional behavioral experiments. Naturalistic multimodal domains with rich underlying semantics (e.g., videos, or musical excerpts; Marjeh et al., 2023; Margulis et al., 2022) are a good example of this: Modern human-in-the-loop paradigms (e.g., STEP-Tag; Marjeh et al., 2023) provide effective ways for deriving semantic taxonomies akin to WordNet from a given set of stimuli. Such taxonomies can then be used to specify a generative process over a semantic hierarchy as in the ImageNet domain.

Specifying a Process Model

In the present work we were mostly concerned with the structure of generative similarity and the way it can be used to derive human-like representations. However, a

follow-up question to address is that of the process by which humans learn to encode generative similarity given the latent structure of the environment, and the way their representations can support the relevant Bayesian computations. Our Monte Carlo formulation (Equations 33 - 34) provides a clear avenue for a process model. It suggests that a learner that is given a stream of “same - different” triplets sampled from the environment, and who seeks to encode “same” stimuli close together, and “different” stimuli further apart, will eventually encode the structure of generative similarity in its representation (see also Appendices E - G). This idea is further supported by a growing line of work on contrastive representations that demonstrates how these can encode probability density ratios such as those used to define generative similarity (i.e., $p(x, y)/p(x)p(y)$), and the way they implement Bayesian inference (Che & Eysenbach, 2025; Eysenbach et al., 2024; Ma & Collins, 2018; Poole et al., 2019). Future work could more closely explore the link between this literature and generative-contrastive learning as a process model.

Accounting for Context Effects

Another question that we did not explore is that of context effects on similarity (Giallanza et al., 2025; Goldstone et al., 1997; Rogers & McClelland, 2004; Rumelhart & Todd, 1993) and the way they can influence the kinds of generative models that people consider. One interesting manifestation of this can be seen in the natural images domain. On the one hand, we found that pairwise similarity judgments are best predicted by a model that is trained on the lowest level of the WordNet hierarchy (i.e., a generative process in which ImageNet classes are sampled and then images are generated from them). However, when modeling human judgments in a triplet odd-one-out task that spanned multiple classes, this model underperformed the more complex one that takes advantage of the full WordNet hierarchy. This suggests that depending on task demands, participants may consider different levels of generative complexity, e.g., by modulating the weights they assign to the different branches (or levels) in the semantic hierarchy (possibly akin to selective attention; Nosofsky, 1992), or by restructuring their generative representation

based on an inferred context (similar to connectionist models of semantic cognition; Giallanza et al., 2025; Rogers and McClelland, 2004), which could be a meaningful target for future modeling work.

Extending to Other Modalities

Although varied, our domains were limited to the visual modality as it is the most widely studied. However, our generative-contrastive approach can be equally applied to other modalities. Indeed, contrastive techniques have been successfully deployed in numerous domains such as audio, video, and language (Gao et al., 2021; Oord et al., 2018; Radford et al., 2021). Moreover, semantic hierarchies, as well as probabilistic grammars similar to those studied in the present work, are also available for many domains (e.g., video: Caba Heilbron et al., 2015; audio: Gemmeke et al., 2017; language: McCoy and Griffiths, 2023; Yang and Piantadosi, 2022). By combining the two, it should be possible to extend the application of generative similarity to other modalities.

Extending to Other Populations

Finally, we note that our participant cohort was restricted to Western populations. An exciting avenue for future work could be to explore how generative assumptions vary cross-culturally, and how that may impact the resulting generative similarity. This is particularly promising in the context of semantic organization underlying rich sensory domains such as music where the semantics are known to vary across cultures (Margulis et al., 2022). Combining generative similarity with semantic mining techniques such as STEP-Tag (Marjeh et al., 2023) could provide a way to explore this space.

Conclusion

Generative processes provide a strong organizing principle underlying complex and changing sensory environments. By reasoning about the processes that produced the stimuli around them, humans form effective representations of the world that support generalization and abstraction. By formulating this idea using Bayesian inference, we showed how foundational and at times conflicting notions of psychological similarity can be

1108 unified under a single formalism and differentiated by the assumptions they make about
1109 how stimuli are generated.

1110 **Acknowledgments**

1111 This work was supported by grant N00014-23-1-2510 from the Office of Naval
1112 Research. S.K. was supported by a Google PhD Fellowship. We thank Mathias
1113 Sable-Meyer for assisting us with accessing the data in his work and general advice.

References

- Artin, E. (2015). *The gamma function*. Courier Dover Publications.
- Battleday, R. M., Peterson, J. C., & Griffiths, T. L. (2020). Capturing human categorization of natural images by combining deep networks and cognitive models. *Nature Communications*, 11(1), 5418.
- Brown, W. (1910). Some experimental results in the correlation of mental abilities 1. *British Journal of Psychology*, 1904-1920, 3(3), 296–322.
- Caba Heilbron, F., Escorcia, V., Ghanem, B., & Carlos Niebles, J. (2015). Activitynet: A large-scale video benchmark for human activity understanding. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 961–970.
- Casella, G., & Berger, R. (2024). *Statistical inference*. CRC press.
- Chater, N., & Vitányi, P. M. (2003). The generalized universal law of generalization. *Journal of Mathematical Psychology*, 47(3), 346–369.
- Che, Y., & Eysenbach, B. (2025). The “law” of the unconscious contrastive learner: Probabilistic alignment of unpaired modalities. *The Thirteenth International Conference on Learning Representations*.
- Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). A simple framework for contrastive learning of visual representations. *International Conference on Machine Learning*, 1597–1607.
- Cheng, K. (2000). Shepard’s universal law supported by honeybees in spatial generalization. *Psychological Science*, 11(5), 403–408.
- Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3), 273–297.
- Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. *IEEE Conference on Computer Vision and Pattern Recognition*, 248–255.

- 1140 Destler, N., Singh, M., & Feldman, J. (2023). Skeleton-based shape similarity. *Psychological*
1141 *review*, 130(6), 1653.
- 1142 Ennis, D. M., Palen, J. J., & Mullen, K. (1988). A multidimensional stochastic theory of
1143 similarity. *Journal of Mathematical Psychology*, 32(4), 449–465.
- 1144 Eysenbach, B., Myers, V., Salakhutdinov, R., & Levine, S. (2024). Inference via
1145 interpolation: Contrastive representations provably enable planning and inference.
1146 *Advances in Neural Information Processing Systems*, 37, 58901–58928.
- 1147 Fellbaum, C. (1998). *Wordnet: An electronic lexical database*. MIT press.
- 1148 Gao, T., Yao, X., & Chen, D. (2021). SimCSE: Simple contrastive learning of sentence
1149 embeddings. *arXiv preprint arXiv:2104.08821*.
- 1150 Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (1995). *Bayesian data analysis*.
1151 Chapman; Hall/CRC.
- 1152 Gemmeke, J. F., Ellis, D. P., Freedman, D., Jansen, A., Lawrence, W., Moore, R. C.,
1153 Plakal, M., & Ritter, M. (2017). Audio set: An ontology and human-labeled dataset
1154 for audio events. *2017 IEEE International Conference on Acoustics, Speech and*
1155 *Signal Processing (ICASSP)*, 776–780.
- 1156 Giallanza, T., Rogers, T. T., & Cohen, J. D. (2025). An integrated model of semantics and
1157 control, part 2: Solving the similarity paradox through context inference.
- 1158 Goldstone, R. L., Medin, D. L., & Halberstadt, J. (1997). Similarity in context. *Memory &*
1159 *Cognition*, 25(2), 237–255.
- 1160 Griffiths, T. L. (2020). Understanding human intelligence through human limitations.
1161 *Trends in Cognitive Sciences*, 24(11), 873–883.
- 1162 Griffiths, T. L., Steyvers, M., & Tenenbaum, J. B. (2007). Topics in semantic
1163 representation. *Psychological review*, 114(2), 211.
- 1164 Griffiths, T. L., & Tenenbaum, J. B. (2007). From mere coincidences to meaningful
1165 discoveries. *Cognition*, 103(2), 180–226.

- 1166 Hall, B. C. (2013). Lie groups, lie algebras, and representations. In *Quantum theory for*
1167 *mathematicians* (pp. 333–366). Springer.
- 1168 He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition.
1169 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*,
1170 770–778.
- 1171 He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep residual learning for image recognition.
1172 arxiv e-prints. *arXiv preprint arXiv:1512.03385*, 10.
- 1173 Hebart, M. N., Zheng, C. Y., Pereira, F., & Baker, C. I. (2020). Revealing the
1174 multidimensional mental representations of natural objects underlying human
1175 similarity judgements. *Nature Human Behaviour*, 4(11), 1173–1185.
- 1176 Henshilwood, C. S., d’Errico, F., Van Niekerk, K. L., Coquinot, Y., Jacobs, Z.,
1177 Lauritzen, S.-E., Menu, M., & García-Moreno, R. (2011). A 100,000-year-old
1178 ochre-processing workshop at Blombos Cave, South Africa. *Science*, 334(6053),
1179 219–222.
- 1180 Hsu, J., Wu, J., & Goodman, N. (2022). Geoclidean: Few-shot generalization in Euclidean
1181 geometry. *Advances in Neural Information Processing Systems*, 35, 39007–39019.
- 1182 Jacoby, N., Undurraga, E. A., McPherson, M. J., Valdés, J., Ossandón, T., &
1183 McDermott, J. H. (2019). Universal and non-universal features of musical pitch
1184 perception revealed by singing. *Current Biology*, 29(19), 3229–3243.
- 1185 James, W. (1890). The principles of psychology. *Henry Holt*.
- 1186 Jha, A., Peterson, J. C., & Griffiths, T. L. (2023). Extracting low-dimensional
1187 psychological representations from convolutional neural networks. *Cognitive Science*,
1188 47(1), e13226.
- 1189 Jones, M. N., Gruenenfelder, T. M., & Recchia, G. (2018). In defense of spatial models of
1190 semantic representation. *New Ideas in Psychology*, 50, 54–60.

- 1191 Kemp, C., Bernstein, A., & Tenenbaum, J. B. (2005). A generative theory of similarity.
1192 *Proceedings of the 27th Annual Conference of the Cognitive Science Society*,
1193 1132–1137.
- 1194 Kemp, C., & Tenenbaum, J. B. (2009). Structured statistical models of inductive
1195 reasoning. *Psychological Review*, 116(1), 20.
- 1196 Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C.,
1197 & Krishnan, D. (2020). Supervised contrastive learning. *Advances in Neural*
1198 *Information Processing Systems*, 33, 18661–18673.
- 1199 Krumhansl, C. L. (1978). Concerning the applicability of geometric models to similarity
1200 data: The interrelationship between similarity and spatial density.
- 1201 Kubilius, J., Schrimpf, M., Kar, K., Rajalingham, R., Hong, H., Majaj, N., Issa, E.,
1202 Bashivan, P., Prescott-Roy, J., Schmidt, K., et al. (2019). Brain-like object
1203 recognition with high-performing shallow recurrent ANNs. *Advances in Neural*
1204 *Information Processing Systems*, 32.
- 1205 Laforgia, A., & Natalini, P. (2012). On the asymptotic expansion of a ratio of gamma
1206 functions. *Journal of Mathematical Analysis and Applications*, 389(2), 833–837.
- 1207 Ma, Z., & Collins, M. (2018). Noise contrastive estimation and negative sampling for
1208 conditional models: Consistency and statistical efficiency. *arXiv preprint*
1209 *arXiv:1809.01812*.
- 1210 Mahalanobis, P. C. (2018). On the generalized distance in statistics. *Sankhyā: The Indian*
1211 *Journal of Statistics, Series A (2008-)*, 80, S1–S7.
- 1212 Margulis, E. H., Wong, P. C., Turnbull, C., Kubit, B. M., & McAuley, J. D. (2022).
1213 Narratives imagined in response to instrumental music reveal culture-bounded
1214 intersubjectivity. *Proceedings of the National Academy of Sciences*, 119(4),
1215 e2110406119.

- 1216 Marjeh, R., Jacoby, N., Peterson, J. C., & Griffiths, T. L. (2024). The universal law of
1217 generalization holds for naturalistic stimuli. *Journal of Experimental Psychology:*
1218 *General*, 153(3), 573.
- 1219 Marjeh, R., Van Rijn, P., Sucholutsky, I., Sumers, T., Lee, H., Griffiths, T. L., &
1220 Jacoby, N. (2023). Words are all you need? Language as an approximation for
1221 human similarity judgments. *The Eleventh International Conference on Learning*
1222 *Representations*.
- 1223 McCoy, R. T., & Griffiths, T. L. (2023). Modeling rapid language learning by distilling
1224 bayesian priors into artificial neural networks. *arXiv preprint arXiv:2305.14701*.
- 1225 McKinley, S. C., & Nosofsky, R. (1995). Investigations of exemplar and decision bound
1226 models in large, ill-defined category structures. *Journal of Experimental Psychology:*
1227 *Human Perception and Performance*, 21(1), 128.
- 1228 Medin, D. L., Goldstone, R. L., & Gentner, D. (1993). Respects for similarity. *Psychological*
1229 *Review*, 100(2), 254.
- 1230 Medin, D. L., & Schaffer, M. M. (1978). Context theory of classification learning.
1231 *Psychological Review*, 85(3), 207.
- 1232 Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed
1233 representations of words and phrases and their compositionality. *Advances in Neural*
1234 *Information Processing Systems*, 26.
- 1235 Miller, G. A. (1995). WordNet: A lexical database for English. *Communications of the*
1236 *ACM*, 38(11), 39–41.
- 1237 Minka, T. (1998). Inferring a Gaussian distribution. *Media Lab Note*.
- 1238 Moschitti, A. (2006). Making tree kernels practical for natural language learning. *11th*
1239 *conference of the European Chapter of the Association for Computational*
1240 *Linguistics*, 113–120.
- 1241 Murphy, G. (2004). *The big book of concepts*. MIT press.

- 1242 Muttenthaler, L., Greff, K., Born, F., Spitzer, B., Kornblith, S., Mozer, M. C.,
1243 Müller, K.-R., Unterthiner, T., & Lampinen, A. K. (2025). Aligning machine and
1244 human visual representations across abstraction levels. *Nature*, 647(8089), 349–355.
- 1245 Muttenthaler, L., Linhardt, L., Dippel, J., Vandermeulen, R. A., Hermann, K.,
1246 Lampinen, A., & Kornblith, S. (2024). Improving neural network representations
1247 using human similarity judgments. *Advances in Neural Information Processing*
1248 *Systems*, 36.
- 1249 Myung, I. J., & Shepard, R. N. (1996). Maximum entropy inference and stimulus
1250 generalization. *Journal of Mathematical Psychology*, 40(4), 342–347.
- 1251 Nosofsky, R. (1986). Attention, similarity, and the identification–categorization
1252 relationship. *Journal of experimental psychology: General*, 115(1), 39.
- 1253 Nosofsky, R. (1985). Overall similarity and the identification of separable-dimension
1254 stimuli: A choice model analysis. *Perception & Psychophysics*, 38, 415–432.
- 1255 Nosofsky, R. (1992). Similarity scaling and cognitive process models. *Annual review of*
1256 *Psychology*, 43(1), 25–53.
- 1257 Oaksford, M., & Chater, N. (1994). A rational analysis of the selection task as optimal
1258 data selection. *Psychological review*, 101(4), 608.
- 1259 Oord, A. v. d., Li, Y., & Vinyals, O. (2018). Representation learning with contrastive
1260 predictive coding. *arXiv preprint arXiv:1807.03748*.
- 1261 Osherson, D. N., Smith, E. E., Wilkie, O., Lopez, A., & Shafir, E. (1990). Category-based
1262 induction. *Psychological review*, 97(2), 185.
- 1263 Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O.,
1264 Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A.,
1265 Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn:
1266 Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.
- 1267 Petersen, K. B., Pedersen, M. S., et al. (2008). The matrix cookbook. *Technical University*
1268 *of Denmark*, 7(15), 510.

- 1269 Peterson, J. C., Abbott, J. T., & Griffiths, T. L. (2018). Evaluating (and improving) the
1270 correspondence between deep neural networks and human representations. *Cognitive*
1271 *Science*, 42(8), 2648–2669.
- 1272 Peterson, J. C., Chen, D., & Griffiths, T. L. (2020). Parallelograms revisited: Exploring the
1273 limitations of vector space models for simple analogies. *Cognition*, 205, 104440.
- 1274 Poole, B., Ozair, S., Van Den Oord, A., Alemi, A., & Tucker, G. (2019). On variational
1275 bounds of mutual information. *International Conference on Machine Learning*,
1276 5171–5180.
- 1277 Posner, M. I., & Keele, S. W. (1968). On the genesis of abstract ideas. *Journal of*
1278 *experimental psychology*, 77(3p1), 353.
- 1279 Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G.,
1280 Askell, A., Mishkin, P., Clark, J., et al. (2021). Learning transferable visual models
1281 from natural language supervision. *International Conference on Machine Learning*,
1282 8748–8763.
- 1283 Reed, S. K. (1972). Pattern recognition and categorization. *Cognitive psychology*, 3(3),
1284 382–407.
- 1285 Rogers, T. T., & McClelland, J. L. (2004). *Semantic cognition: A parallel distributed*
1286 *processing approach*. MIT Press.
- 1287 Rumelhart, D. E., & Todd, P. M. (1993). Learning and connectionist representations. In
1288 D. E. Meyer & S. Kornblum (Eds.), *Attention and performance xiv: Synergies in*
1289 *experimental psychology, artificial intelligence, and cognitive neuroscience*
1290 (pp. 3–30). MIT Press.
- 1291 Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z.,
1292 Karpathy, A., Khosla, A., Bernstein, M., et al. (2015). Imagenet large scale visual
1293 recognition challenge. *International Journal of Computer Vision*, 115, 211–252.
- 1294 Sablé-Meyer, M., Fagot, J., Caparos, S., van Kerkoerle, T., Amalric, M., & Dehaene, S.
1295 (2021). Sensitivity to geometric shape regularity in humans and baboons: A

putative signature of human singularity. *Proceedings of the National Academy of Sciences*, 118(16), e2023123118.

Saito, A., Hayashi, M., Takeshita, H., & Matsuzawa, T. (2014). The origin of representational drawing: A comparison of human children and chimpanzees. *Child Development*, 85(6), 2232–2246.

Santini, S., & Jain, R. (1999). Similarity measures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(9), 871–883.

Schrimpf, M., Kumbhani, J., Hong, H., Majaj, N. J., Rajalingham, R., Issa, E. B., Kar, K., Bashivan, P., Prescott-Roy, J., Geiger, F., et al. (2018). Brain-score: Which artificial neural network for object recognition is most brain-like? *BioRxiv*, 407007.

Shepard, R. N. (1980). Multidimensional scaling, tree-fitting, and clustering. *Science*, 210(4468), 390–398.

Shepard, R. N. (1987). Toward a universal law of generalization for psychological science. *Science*, 237(4820), 1317–1323.

Shepard, R. N., & Arabie, P. (1979). Additive clustering: Representation of similarities as combinations of discrete overlapping properties. *Psychological Review*, 86(2), 87.

Sims, C. R. (2018). Efficient coding explains the universal law of generalization in human perception. *Science*, 360(6389), 652–656.

Simson, R. (1838). *The elements of Euclid*. Desilver, Thomas.

Sohn, K. (2016). Improved deep metric learning with multi-class n-pair loss objective. *Advances in Neural Information Processing Systems*, 29.

Sucholutsky, I., Muttenthaler, L., Weller, A., Peng, A., Bobu, A., Kim, B., Love, B. C., Grant, E., Achterberg, J., Tenenbaum, J. B., et al. (2023). Getting aligned on representational alignment. *arXiv preprint arXiv:2310.13018*.

Tenenbaum, J. B. (1999). Rules and similarity in concept learning. *Advances in Neural Information Processing Systems*, 12.

- 1322 Tenenbaum, J. B., & Griffiths, T. L. (2001). Generalization, similarity, and Bayesian
1323 inference. *Behavioral and Brain Sciences*, *24*(4), 629–640.
- 1324 Tenenbaum, J. B., Kemp, C., Griffiths, T. L., & Goodman, N. D. (2011). How to grow a
1325 mind: Statistics, structure, and abstraction. *Science*, *331*(6022), 1279–1285.
- 1326 Tversky, A. (1977). Features of similarity. *Psychological Review*, *84*(4), 327.
- 1327 Tversky, A., & Hutchinson, J. (1986). Nearest neighbor analysis of psychological spaces.
1328 *Psychological Review*, *93*(1), 3.
- 1329 Winawer, J., Witthoft, N., Frank, M. C., Wu, L., Wade, A. R., & Boroditsky, L. (2007).
1330 Russian blues reveal effects of language on color discrimination. *Proceedings of the*
1331 *National Academy of Sciences*, *104*(19), 7780–7785.
- 1332 Yang, Y., & Piantadosi, S. T. (2022). One model for the learning of language. *Proceedings*
1333 *of the National Academy of Sciences*, *119*(5), e2021865119.
- 1334 Zaslavsky, N., Kemp, C., Regier, T., & Tishby, N. (2018). Efficient compression in color
1335 naming and its evolution. *Proceedings of the National Academy of Sciences*,
1336 *115*(31), 7937–7942.

Appendix A

Proof of Convexity

Our goal is to show that the Bayesian similarity measure of Tenenbaum and Griffiths (2001), i.e., $s_g(x, y) = s_0 \sum_{h:x,y \in h} \frac{1}{|h|} \pi(|h|)$, as well as our measure of generative similarity $s_{gen}(x, y) = s_0^2 \sum_{h:x,y \in h} \frac{1}{|h|^2} \pi(|h|)$ both yield monotonically decreasing convex functions of the separation between x and y . To do so, we will show that any similarity function of the form $s(x, y) = \sum_{h:x,y \in h} f(|h|)$ where $f(|h|)$ is some non-negative function (and strictly positive for at least some value $|h_0|$ since $\pi(|h|)$ is a probability distribution that must sum to 1) for which the sum s is finite and h are convex (i.e., if $x, y \in h$ then the line connecting them is also in h) satisfies this requirement. We will show this for a single dimension first, but equivalent arguments can be made in higher dimensions (see below) by generalizing consequential intervals into higher-dimensional regions and choosing an appropriate coordinate system.

In analogy to Tenenbaum and Griffiths (2001), let us consider a one-dimensional line $\Lambda(\Delta)$ discretized to small units $0 < \Delta \ll 1$ (Figure A1A). To show that s is decreasing, we need to prove that $s(x, y) \geq s(x, y + \Delta)$ (with strict inequality for at least some value). Without loss of generality let us assume that $x < y$. In this case, we have

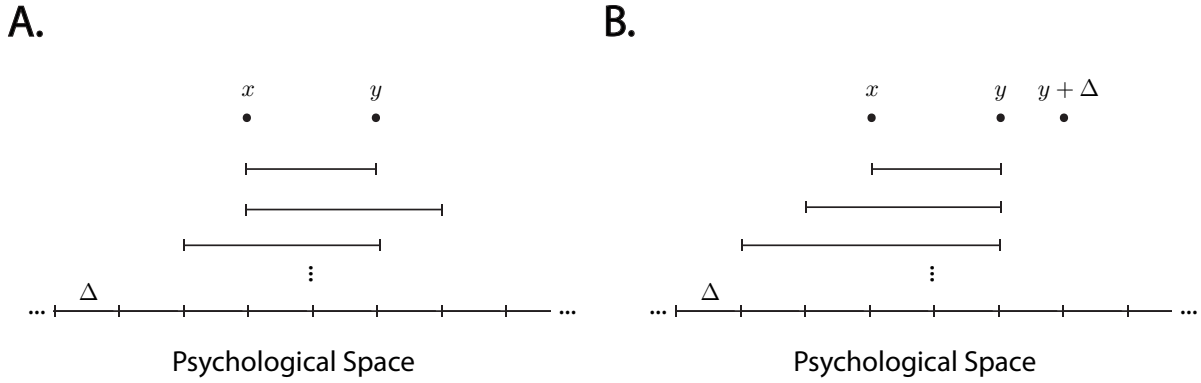
$$s(x, y) - s(x, y + \Delta) = \sum_{h:x,y \in h} f(|h|) - \sum_{h:x,y+\Delta \in h} f(|h|) \quad (\text{A1})$$

Now, since by definition x, y are contained in the interval $[x, y + \Delta]$, it follows that any convex consequential region h containing $x, y + \Delta$ (and hence $[x, y + \Delta]$) must also contain x, y but not vice versa. This means that we can write

$$\sum_{h:x,y \in h} f(|h|) = \sum_{h:x,y+\Delta \in h} f(|h|) + R_\Delta \quad (\text{A2})$$

where $R_\Delta \geq 0$ is some non-negative residue because $f(|h|)$ is non-negative. From here it follows that

$$s(x, y) - s(x, y + \Delta) = R_\Delta \geq 0 \quad (\text{A3})$$

**Figure A1**

*Schematic of the one-dimensional proof construction. **A.** Discretized one-dimensional psychological space (in units of Δ) along with example consequential regions that jointly contain stimuli x and y . **B.** Consequential intervals contributing to the residue R_Δ .*

1357 Since this holds for any y and $f(|h|)$ is strictly positive for at least some value, we conclude
 1358 that $s(x, y)$ is decreasing as desired. Next, to show that $s(x, y)$ is convex, we need to show
 1359 that it is decelerating (i.e., it has a non-negative second derivative). In finite difference
 1360 form, this amounts to showing that $s(x, y) - s(x, y + \Delta)$ is never smaller than
 1361 $s(x, y + \Delta) - s(x, y + 2\Delta)$ (and strictly larger for some values). If we denote $|y - x| = n\Delta$
 1362 for some integer $n > 0$, then observe that the residue R_Δ (i.e., the difference between the
 1363 terms of $s(x, y)$ and $s(x, y + \Delta)$) corresponds to all intervals that contain x, y but not
 1364 $y + \Delta$. These are tantamount to all consequential intervals h of length $n\Delta$ or larger that
 1365 end on y and contain x (if they extended beyond y then they contain $y + \Delta$ and hence
 1366 cancel out in the difference; Figure A1B). Thus,

$$s(x, y) - s(x, y + \Delta) = f(n\Delta) + f((n + 1)\Delta) + \dots \quad (\text{A4})$$

1367 Likewise, we have

$$s(x, y + \Delta) - s(x, y + 2\Delta) = f((n + 1)\Delta) + f((n + 2)\Delta) + \dots \quad (\text{A5})$$

1368 From which it follows that

$$[s(x, y) - s(x, y + \Delta)] - [s(x, y + \Delta) - s(x, y + 2\Delta)] = f(n\Delta) \geq 0 \quad (\text{A6})$$

1369 and since f is strictly positive for at least some values, we arrive at the desired result.

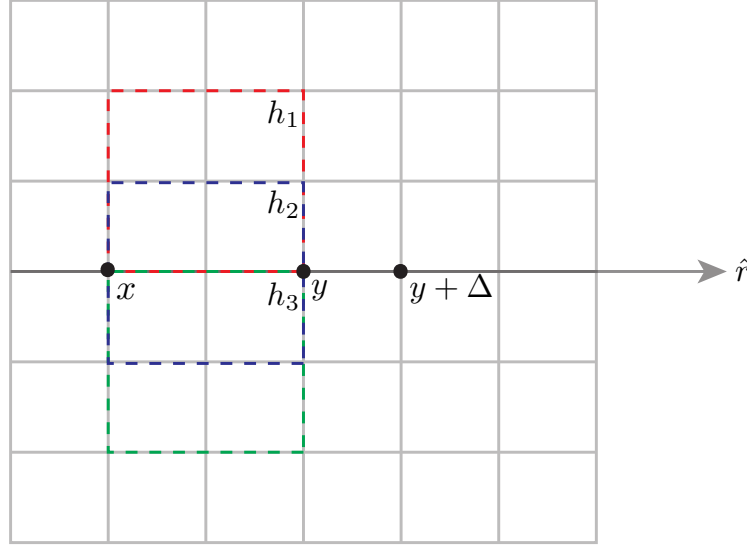


Figure A2

Two-dimensional construction. The regions h_1, h_2, h_3 are the smallest consequential regions containing both x and y .

1370 To generalize this analysis to higher dimensions, we choose a coordinate system in
 1371 which one of the axes is along the line connecting the points x, y and work with a finely
 1372 discretized multi-dimensional lattice $\Lambda(\Delta)$ (Figure A2). We will show the proof first for a
 1373 natural extension of consequential intervals to consequential hypercubes along the lattice
 1374 axes (in two dimensions these are just squares which satisfy the convex set assumption),
 1375 and then provide a more general proof.

1376 Our goal is to show that along the axis connecting x and y , which we denote by the
 1377 unit vector $\hat{r}$, the similarity function is decreasing and decelerating. Showing that it is

decreasing is straightforward. Since y is along the line connecting x and $y + \Delta\hat{r}$, any convex consequential region connecting x and $y + \Delta\hat{r}$ must also contain y , from which it follows as in the one dimensional case that $s(x, y) \geq s(x, y + \Delta\hat{r})$ (the strict inequality for some cases due to the positivity properties of f remains unchanged here and below). As for the deceleration property, we need to show that the residue $s(x, y) - s(x, y + \Delta\hat{r})$ is larger than the residue $s(x, y + \Delta\hat{r}) - s(x, y + 2\Delta\hat{r})$. We will show this explicitly in two-dimensions and provide a more general proof afterwards. Let us denote the separation between x and y along $\hat{r}$ as $n\Delta$. Next, observe that the residue $s(x, y) - s(x, y + \Delta\hat{r})$ follows from all the squares that contain x and y but not $y + \Delta\hat{r}$ (if they did contain $y + \Delta\hat{r}$, then from set convexity they would also contain y and cancel out between the two terms). These are all the squares whose right side contains y , and there are $m + 1$ such squares for a region of lateral width $m\Delta$ (see Figure A2 for illustration). Note that the smallest possible width of a square containing x and y is $n\Delta$. Thus, we can write

$$s(x, y) - s(x, y + \Delta\hat{r}) = (n + 1)f((n\Delta)^2) + (n + 2)f(((n + 1)\Delta)^2) + \dots \quad (\text{A7})$$

and similarly for the second residue

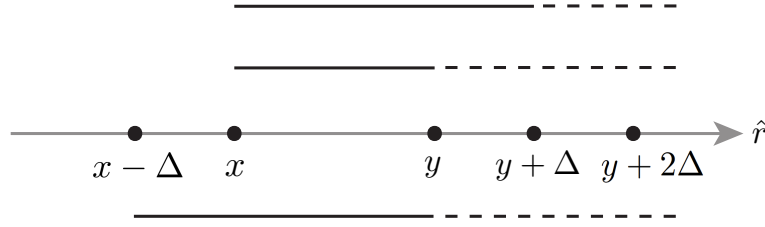
$$s(x, y + \Delta\hat{r}) - s(x, y + 2\Delta\hat{r}) = (n + 2)f(((n + 1)\Delta)^2) + (n + 3)f(((n + 2)\Delta)^2) + \dots \quad (\text{A8})$$

from which we conclude that

$$[s(x, y) - s(x, y + \Delta\hat{r})] - [s(x, y + \Delta\hat{r}) - s(x, y + 2\Delta\hat{r})] = (n + 1)f((n\Delta)^2) \geq 0 \quad (\text{A9})$$

which is the desired result.

To prove this result more generally for any number of dimensions, we will rely on the properties of convex sets. First, note that the proof for $s(x, y) \geq s(x, y + \Delta\hat{r})$ remains unchanged: any convex consequential region containing x and $y + \Delta\hat{r}$ must contain the line connecting them, and hence x, y . As for the deceleration property, observe that the set of consequential regions contributing to the residue $s(x, y) - s(x, y + \Delta\hat{r})$ corresponds to all convex sets h that contain the line segment connecting x, y , but not any point along its

**Figure A3**

The three line segments used in the general proof construction. Solid line indicates the part of the $\hat{r}$ line that should be included, and the dashed line indicates the part that must be excluded.

1400 extension along $y + \Delta\hat{r}$ (if it did, then from convexity it must contain $y + \Delta\hat{r}$ and thus
 1401 cancel out; Figure A3). Since all terms depend on the size of the consequential region but
 1402 not on its location, the result is identical for any line segment of size $\|x - y\| = n\Delta$ (i.e.,
 1403 summing over all consequential regions that contain a line segment of length $n\Delta$ but not its
 1404 extension along one direction), and so we denote the outcome of this sum by $F(n\Delta)$. Next,
 1405 we can make a similar argument for $s(x, y + \Delta\hat{r}) - s(x, y + 2\Delta\hat{r})$, with the only difference
 1406 being that the length of the line segment is now $\|x - (y + \Delta\hat{r})\| = (n + 1)\Delta$ and so we
 1407 denote the outcome of the sum by $F((n + 1)\Delta)$. Crucially, observe that any consequential
 1408 region containing the line connecting $x - \Delta\hat{r}$ to y but not its extension beyond y makes a
 1409 valid contribution to $s(x, y) - s(x, y + \Delta\hat{r})$ (because it also contains x from convexity and
 1410 excludes $y + \Delta\hat{r}$; Figure A3), but since this line segment is of length $(n + 1)\Delta$, it makes an
 1411 identical contribution to $s(x, y + \Delta\hat{r}) - s(x, y + 2\Delta\hat{r})$, i.e., $F((n + 1)\Delta)$. From here it
 1412 follows that $F(n\Delta) = F((n + 1)\Delta) + r_{n\Delta} \geq F((n + 1)\Delta)$ (the residue $r_{n\Delta}$ corresponding to
 1413 all consequential regions containing x, y but neither of its line extensions beyond x or y ,
 1414 consistent with Equations A6 and A9). This completes the proof.

Appendix B

Sampling From an Indefinite Gaussian

In what follows, we reproduce the proof of Griffiths and Tenenbaum (2007) for the probability of a set of points $d = \{x_1, \dots, x_n\}$ under the assumption that they were sampled from the same indefinite Gaussian $\mathcal{N}(\mu, \Sigma)$ given a uniform prior over the means for some large region R of size $|R|$, and an inverse Wishart prior with hyperparameters $k \geq D$ and I where I is the identity matrix

$$p(\mu) = \frac{1}{|R|}, \quad p(\Sigma) = \frac{1}{c_{kD} |\Sigma|^{(k+D+1)/2}} \exp \left[-\frac{1}{2} \text{tr}(\Sigma^{-1}) \right]. \quad (\text{B1})$$

where $c_{kD} > 0$ is a constant that depends on the hyperparameter k and the dimensionality D . To compute $p(d|\text{Gaussian})$, we need to sum over all possible values of μ and Σ . To do so, observe first that

$$\begin{aligned}
 p(d|\mu, \Sigma) &= \frac{1}{(2\pi)^{Dn/2} |\Sigma|^{n/2}} \exp \left(-\frac{1}{2} \sum_i (x_i - \mu)^\top \Sigma^{-1} (x_i - \mu) \right) \\
 &= \frac{1}{(2\pi)^{Dn/2} |\Sigma|^{n/2}} \exp \left(-\frac{n}{2} (\bar{x} - \mu)^\top \Sigma^{-1} (\bar{x} - \mu) - \frac{1}{2} \text{tr}(S \Sigma^{-1}) \right)
 \end{aligned}$$

where $\top$ is the transpose operation, $\bar{x} = \frac{1}{n} \sum_i x_i$, and $S = \sum_i (x_i - \bar{x})(x_i - \bar{x})^\top$. The second equality follows from the observation

$$\begin{aligned}
 \text{tr}(S \Sigma^{-1}) &= \sum_{i,j,m} \left(x_{i(j)} - \frac{1}{n} \sum_k x_{k(j)} \right) \left(x_{i(m)} - \frac{1}{n} \sum_k x_{k(m)} \right) (\Sigma^{-1})_{(mj)} \\
 &= \sum_{i,j,m} \left(x_{i(m)} - \frac{1}{n} \sum_k x_{k(m)} \right) (\Sigma^{-1})_{(mj)} \left(x_{i(j)} - \frac{1}{n} \sum_k x_{k(j)} \right) \\
 &= \sum_i (x_i - \bar{x})^\top \Sigma^{-1} (x_i - \bar{x}) \\
 &= \sum_i (x_i - \mu + \mu - \bar{x})^\top \Sigma^{-1} (x_i - \mu + \mu - \bar{x}) \\
 &= \sum_i (x_i - \mu)^\top \Sigma^{-1} (x_i - \mu) + n(\bar{x} - \mu)^\top \Sigma^{-1} (\bar{x} - \mu) \\
 &\quad + \sum_i (x_i - \mu)^\top \Sigma^{-1} (\mu - \bar{x}) + \sum_i (\mu - \bar{x})^\top \Sigma^{-1} (x_i - \mu)
 \end{aligned}$$

1425 Now, note that

$$\begin{aligned} \sum_i (x_i - \mu)^\top \Sigma^{-1} (\mu - \bar{x}) &= \left(\sum_i x_i^\top - \sum_i \mu^\top \right) \Sigma^{-1} (\mu - \bar{x}) \\ &= (n\bar{x}^\top - n\mu^\top) \Sigma^{-1} (\mu - \bar{x}) \\ &= -n(\bar{x} - \mu) \Sigma^{-1} (\bar{x} - \mu) \end{aligned}$$

1426 and that the same holds for $\sum_i (\mu - \bar{x})^\top \Sigma^{-1} (x_i - \mu)$, from which we conclude that

$$\text{tr}(S\Sigma^{-1}) = \sum_i (x_i - \mu)^\top \Sigma^{-1} (x_i - \mu) - n(\bar{x} - \mu)^\top \Sigma^{-1} (\bar{x} - \mu) \quad (\text{B2})$$

1427 which is the desired result. From here, we can proceed to the integration over all Gaussian
1428 parameters

$$\begin{aligned} p(d|\text{Gaussian}) &= \int p(d|\mu, \Sigma) p(\mu) p(\Sigma) d\mu d\Sigma \\ &= \frac{1}{|R|(2\pi)^{Dn/2} c_{kD}} \int \frac{1}{|\Sigma|^{(k+D+1+n)/2}} \exp\left(-\frac{1}{2}\text{tr}((S+I)\Sigma^{-1})\right) d\Sigma \\ &\quad \times \int \exp\left(-\frac{n}{2}(\bar{x} - \mu)^\top \Sigma^{-1} (\bar{x} - \mu)\right) d\mu \end{aligned}$$

1429 Now, note that the integral over μ is simply that of an (unnormalized) multivariate
1430 Gaussian distribution with covariance matrix Σ/n and mean $\bar{x}$. As long as the range R is
1431 sufficiently large relative to Σ , we can simply approximate it by the integral over all of $\mathbb{R}^D$,
1432 in which case it becomes the normalization constant of the Gaussian $(2\pi)^{D/2} n^{-D/2} |\Sigma|^{1/2}$.
1433 Thus, we have

$$p(d|\text{Gaussian}) \approx \frac{1}{|R|(2\pi)^{D(n-1)/2} n^{D/2} c_{kD}} \int \frac{1}{|\Sigma|^{(k+D+n)/2}} \exp\left(-\frac{1}{2}\text{tr}((S+I)\Sigma^{-1})\right) d\Sigma$$

1434 Finally, note that the remaining integrand has the form of an inverse-Wishart distribution
1435 with hyperparameters $k' = k + n - 1$ and $S + I$, up to normalization factors that are
1436 independent of Σ . Formally, a properly normalized inverse-Wishart distribution with
1437 hyperparameters k' and Ψ is given by (Gelman et al., 1995)

$$p(\Sigma) = \frac{|\Psi|^{k'/2}}{c_{k'D} |\Sigma|^{(k'+D+1)/2}} \exp\left[-\frac{1}{2}\text{tr}(\Psi\Sigma^{-1})\right]. \quad (\text{B3})$$

1438 where as before $c_{k'D} > 0$ is a constant that depends on the hyperparameter k' and the
 1439 dimensionality D . Thus, we conclude that

$$p(d|\text{Gaussian}) \approx \frac{c_{k'D}}{|R|(2\pi)^{D(n-1)/2}n^{D/2}c_{kD}} \frac{1}{|S+I|^{(k+n-1)/2}} \quad (\text{B4})$$

1440 For completeness, we note that $c_{kD} = 2^{Dk/2}\pi^{D(D-1)/4} \prod_{i=1}^D \Gamma\left(\frac{k+1-i}{2}\right)$, where Γ is the
 1441 Gamma function. Using this we can further simply

$$\begin{aligned} \frac{c_{k'D}}{(2\pi)^{D(n-1)/2}n^{D/2}c_{kD}} &= \frac{2^{D(k+n-1)/2}\pi^{D(D-1)/4} \prod_{i=1}^D \Gamma\left(\frac{k+n-i}{2}\right)}{(2\pi)^{D(n-1)/2}n^{D/2}2^{Dk/2}\pi^{D(D-1)/4} \prod_{i=1}^D \Gamma\left(\frac{k+1-i}{2}\right)} \\ &= \frac{1}{\pi^{D(n-1)/2}n^{D/2}} \prod_{i=1}^D \frac{\Gamma\left(\frac{k+n-i}{2}\right)}{\Gamma\left(\frac{k+1-i}{2}\right)} \end{aligned}$$

1442 Thus, we arrive at the final formula of Griffiths and Tenenbaum (2007)

$$p(d|\text{Gaussian}) \approx \frac{\mathcal{C}_{k,D,n}}{|S+I|^{(k+n-1)/2}}, \quad \mathcal{C}_{k,D,n} = \frac{1}{\pi^{D(n-1)/2}n^{D/2}} \prod_{i=1}^D \frac{\Gamma\left(\frac{k+n-i}{2}\right)}{\Gamma\left(\frac{k+1-i}{2}\right)}. \quad (\text{B5})$$

1443 Crucially, note that the only dependence on the exact value of the stimuli d is in the
 1444 determinant term and the rest are simply constant factors.

Appendix C

Deriving Tversky's Model

Our goal is to derive the generative similarity measure associated with the process specified in Equation (16)

$$s(\sigma_1(f^{(1)}), \sigma_2(f^{(2)})) = \frac{p(\text{same}|\sigma_1, \sigma_2)}{p(\text{different}|\sigma_1, \sigma_2)} = \frac{p_{\text{same}}(\sigma_1, \sigma_2)}{p_{\text{diff}}(\sigma_1, \sigma_2)} \quad (\text{C1})$$

where p_{same} and p_{diff} are the likelihoods under the “same” and “different” processes (Figure 1A) and, as before, we assume uniform priors $p(\text{same}) = p(\text{different})$. Plugging in the different Beta, Bernoulli, and uniform distributions, we have

$$p_{\text{same}}(\sigma_1, \sigma_2) = \int \frac{1}{|\mathcal{S}(f^{(1)})|} \frac{1}{|\mathcal{S}(f^{(2)})|} \prod_i \left[\theta_i^{f_i^{(1)} + f_i^{(2)} + \alpha - 1} (1 - \theta_i)^{\bar{f}_i^{(1)} + \bar{f}_i^{(2)} + \beta - 1} / B(\alpha, \beta) \right] d\theta_{1 \dots n}$$

where we defined $\bar{f}_i = 1 - f_i$ and used the definitions $\text{Bern}(f_i; \theta_i) = \theta_i^{f_i} (1 - \theta_i)^{\bar{f}_i}$ and $\text{Beta}(\theta_i; \alpha, \beta) = \theta_i^{\alpha-1} (1 - \theta_i)^{\beta-1} / B(\alpha, \beta)$, where B is the Beta function which is given by $B(z_1, z_2) = \int_0^1 dt t^{z_1-1} (1-t)^{z_2-1}$ for all positive numbers $z_1, z_2 > 0$. Likewise, $|\mathcal{S}(f)|$ is the cardinality of the stimulus set associated with the feature vector f and accounts for uniform sampling. Similarly, for the “different” process we have

$$p_{\text{diff}}(\sigma_1, \sigma_2) = \int \frac{1}{|\mathcal{S}(f^{(1)})|} \prod_i \left[\theta_{(1)i}^{f_{(1)i}^{(1)} + \alpha - 1} (1 - \theta_{(1)i})^{\bar{f}_{(1)i}^{(1)} + \beta - 1} / B(\alpha, \beta) \right] d\theta_{(1)1 \dots n} \\ \times \int \frac{1}{|\mathcal{S}(f^{(2)})|} \prod_j \left[\theta_{(2)j}^{f_{(2)j}^{(2)} + \alpha - 1} (1 - \theta_{(2)j})^{\bar{f}_{(2)j}^{(2)} + \beta - 1} / B(\alpha, \beta) \right] d\theta_{(2)1 \dots n}$$

The above integrals might seem complicated at first but the conjugacy relation between the Beta and Bernoulli distributions simplify things drastically. Indeed, the cardinality factors cancel out in the ratio $p_{\text{same}}/p_{\text{diff}}$ so that we are left with a collection of Beta function factors (see definition of Beta function above)

$$s(\sigma_1(f^{(1)}), \sigma_2(f^{(2)})) = \frac{\prod_i B(f_i^{(1)} + f_i^{(2)} + \alpha, \bar{f}_i^{(1)} + \bar{f}_i^{(2)} + \beta) B(\alpha, \beta)}{\prod_i B(f_i^{(1)} + \alpha, \bar{f}_i^{(1)} + \beta) \prod_j B(f_j^{(2)} + \alpha, \bar{f}_j^{(2)} + \beta)} \quad (\text{C2})$$

Taking the logarithm and rearranging the terms we have

$$\log s(\sigma_1(f^{(1)}), \sigma_2(f^{(2)})) = \sum_i \log \frac{B(f_i^{(1)} + f_i^{(2)} + \alpha, \bar{f}_i^{(1)} + \bar{f}_i^{(2)} + \beta) B(\alpha, \beta)}{B(f_i^{(1)} + \alpha, \bar{f}_i^{(1)} + \beta) B(f_i^{(2)} + \alpha, \bar{f}_i^{(2)} + \beta)} \quad (\text{C3})$$

1460 Now, recall the following Beta function identities¹¹

$$B(x+1, y) = \frac{x}{x+y} B(x, y); \quad B(x, y+1) = \frac{y}{x+y} B(x, y) \quad (C4)$$

1461 Using these identities we can group and simplify the different ratios contributing to the
1462 sum depending on the values of the features. If $f_i^{(1)} = f_i^{(2)} = 1$ then we have

$$\log \frac{B(2+\alpha, 0+\beta)B(\alpha, \beta)}{B(1+\alpha, 0+\beta)B(1+\alpha, 0+\beta)} = \log \frac{\alpha+1}{\alpha+\beta+1} \frac{\alpha+\beta}{\alpha} \quad (C5)$$

1463 If on the other hand $f_i^{(1)} = 1$ and $f_i^{(2)} = 0$ or $f_i^{(1)} = 0$ and $f_i^{(2)} = 1$ then we have

$$\log \frac{B(1+\alpha, 1+\beta)B(\alpha, \beta)}{B(1+\alpha, 0+\beta)B(0+\alpha, 1+\beta)} = \log \frac{\beta}{\alpha+\beta+1} \frac{\alpha+\beta}{\beta} \quad (C6)$$

1464 and finally for $f_i^{(1)} = f_i^{(2)} = 0$ we have¹²

$$\log \frac{B(0+\alpha, 2+\beta)B(\alpha, \beta)}{B(0+\alpha, 1+\beta)B(0+\alpha, 1+\beta)} = \log \frac{\beta+1}{\alpha+\beta+1} \frac{\alpha+\beta}{\beta} \quad (C7)$$

1465 Next, defining Σ_1 and Σ_2 to be the sets of features that hold true for stimuli σ_1 and σ_2 , we
1466 can write

$$\begin{aligned} \log s(\sigma_1(f^{(1)}), \sigma_2(f^{(2)})) &= |\Sigma_1 \cap \Sigma_2| \log \frac{\alpha+1}{\alpha+\beta+1} \frac{\alpha+\beta}{\alpha} \\ &\quad + (|\Sigma_1 - \Sigma_2| + |\Sigma_2 - \Sigma_1|) \log \frac{\beta}{\alpha+\beta+1} \frac{\alpha+\beta}{\beta} \\ &\quad + |\bar{\Sigma}_1 \cap \bar{\Sigma}_2| \log \frac{\beta+1}{\alpha+\beta+1} \frac{\alpha+\beta}{\beta} \end{aligned}$$

1467 where $|\Sigma_1 \cap \Sigma_2|$ is the number features that hold true for both stimuli, $|\Sigma_i - \Sigma_j|$ is the
1468 number of features that hold true for σ_i but not for σ_j , and finally $|\bar{\Sigma}_1 \cap \bar{\Sigma}_2|$ is the number
1469 of features that hold neither for σ_1 nor for σ_2 . Observe next that by definition we have

¹¹ These follow from the fact that $B(z_1, z_2) = \Gamma(z_1)\Gamma(z_2)/\Gamma(z_1+z_2)$ where Γ is the Gamma function which satisfies $\Gamma(z+1) = z\Gamma(z)$ for any $z > 0$ (Artin, 2015).

¹² To the best of our knowledge, the contribution of this condition was not included in the calculation found in Kemp et al. (2005) (see Supplementary of that work).

1470 $|\bar{\Sigma}_1 \cap \bar{\Sigma}_2| = n - |\Sigma_1 \cap \Sigma_2| - |\Sigma_1 - \Sigma_2| - |\Sigma_2 - \Sigma_1|$ where n is the total number of features.

1471 From here it follows that

$$\begin{aligned} \log s(\sigma_1(f^{(1)}), \sigma_2(f^{(2)})) &= |\Sigma_1 \cap \Sigma_2| \log \frac{\alpha+1}{\alpha} \frac{\beta}{\beta+1} \\ &\quad + (|\Sigma_1 - \Sigma_2| + |\Sigma_2 - \Sigma_1|) \log \frac{\beta}{\beta+1} \\ &\quad + n \log \frac{\beta+1}{\alpha+\beta+1} \frac{\alpha+\beta}{\beta}. \end{aligned} \quad (\text{C8})$$

1472 Finally, defining

$$\alpha_s = \log \left(\frac{\alpha+1}{\alpha} \frac{\beta}{\beta+1} \right), \quad \beta_s = \log \left(\frac{\beta+1}{\beta} \right), \quad \log c_0 = n \log \left(\frac{\beta+1}{\alpha+\beta+1} \frac{\alpha+\beta}{\beta} \right) \quad (\text{C9})$$

1473 we arrive at

$$\log s(\sigma_1(f^{(1)}), \sigma_2(f^{(2)})) = \alpha_s |\Sigma_1 \cap \Sigma_2| - \beta_s (|\Sigma_1 - \Sigma_2| + |\Sigma_2 - \Sigma_1|) + \log c_0 \quad (\text{C10})$$

1474 which is the desired result. Finally, to get the alternative formula note that using the

1475 definition of the union $|\Sigma_1 \cup \Sigma_2| = |\Sigma_1 \cap \Sigma_2| + |\Sigma_1 - \Sigma_2| + |\Sigma_2 - \Sigma_1|$ we can rewrite the

1476 formula as

$$\log s(\sigma_1(f^{(1)}), \sigma_2(f^{(2)})) = |\Sigma_1 \cap \Sigma_2| \log \frac{\alpha+1}{\alpha} - |\Sigma_1 \cup \Sigma_2| \log \frac{\beta+1}{\beta} + \log c_0. \quad (\text{C11})$$

Appendix D

Computing the Prototype Limit

1477 To derive Equation (22), we will first consider the case of $D = 1$ dimensions, and then
 1478 generalize to $D > 1$. Throughout, we will assume that as $n \rightarrow \infty$ the set of category
 1479 exemplars $\{x_i\}$ has a finite mean (i.e., $\bar{x}(\{x_i\}) = \frac{1}{n} \sum_{i=1}^n x_i < \infty$) and covariance (i.e.,
 1480 $\Sigma(\{x_i\}) = S(\{x_i\})/n < \infty$; in $D = 1$ this is just the variance $\Sigma(\{x_i\}) = \sigma^2(\{x_i\}) < \infty$).
 1481 Otherwise, this would contradict the Gaussian generative assumption and an alternative,
 1482 more suitable model should be considered.

1483 Starting from the $D = 1$ case, observe that when $n \rightarrow \infty$ the mean of the set
 1484 $\{x_i\} \cup \{x_0\}$ is very close to that of $\{x_i\}$ since adding an extra point to a large set of points
 1485 has a vanishing effect on the mean. Formally, $\frac{1}{n+1} (\sum_{i=1}^n x_i + x_0) = \bar{x}(\{x_i\}) + \mathcal{O}(n^{-1})$ where
 1486 $\mathcal{O}(n^{-1})$ denotes corrections of order n^{-1} or smaller. Next, note that in one dimension we
 1487 have $S(\{x_i\}) = \sum_{i=1}^n (x_i - \bar{x})^2$ from which it follows that

$$S(\{x_i\} \cup \{x_0\}) = S(\{x_i\}) + (x_0 - \bar{x}(\{x_i\}))^2 + \mathcal{O}(n^{-1}) \equiv n\sigma^2(\{x_i\}) + \Delta + r_0/n. \quad (\text{D1})$$

1488 where we defined $\Delta = (x_0 - \bar{x}(\{x_i\}))^2$ and r_0/n is the residue where $r_0 \sim \mathcal{O}(1)$ (since the
 1489 remaining terms are $\mathcal{O}(n^{-1})$). To properly take the limit, we should also take into account
 1490 the dependence on n in the normalization constant that we suppressed in the main text,
 1491 but provide in full in Equation (B5). When computing the ratio between $p_G(\{x_i\} \cup \{x_0\})$
 1492 and $p_G(\{x_i\})$, the normalization factors yield

$$\begin{aligned} \frac{\mathcal{C}_{k,D,n+1}}{\mathcal{C}_{k,D,n}} &= \frac{\pi^{D(n-1)/2} n^{D/2}}{\pi^{D(n+1-1)/2} (n+1)^{D/2}} \prod_{i=1}^D \frac{\Gamma\left(\frac{k+n+1-i}{2}\right)}{\Gamma\left(\frac{k+1-i}{2}\right)} \frac{\Gamma\left(\frac{k+1-i}{2}\right)}{\Gamma\left(\frac{k+n-i}{2}\right)} \\ &= \frac{1}{\pi^{D/2}} \left(\frac{n}{n+1}\right)^{D/2} \prod_{i=1}^D \frac{\Gamma\left(\frac{k+n+1-i}{2}\right)}{\Gamma\left(\frac{k+n-i}{2}\right)} \\ &= \frac{1}{\pi^{D/2}} \left(\frac{n}{n+1}\right)^{D/2} \frac{\Gamma\left(\frac{k+n}{2}\right) \cdots \Gamma\left(\frac{k+n-(D-1)}{2}\right)}{\Gamma\left(\frac{k+n-1}{2}\right) \cdots \Gamma\left(\frac{k+n-D}{2}\right)} \\ &= \frac{1}{\pi^{D/2}} \left(\frac{n}{n+1}\right)^{D/2} \frac{\Gamma\left(\frac{k+n}{2}\right)}{\Gamma\left(\frac{k+n-D}{2}\right)} \end{aligned}$$

When $n \rightarrow \infty$ we can further expand this formula using the asymptotic properties of the Gamma function¹³ (Laforgia & Natalini, 2012)

$$\frac{\mathcal{C}_{k,D,n+1}}{\mathcal{C}_{k,D,n}} = \frac{1}{(2\pi)^{D/2}} n^{D/2} + \mathcal{O}(n^{D/2-1}) \equiv \frac{1}{(2\pi)^{D/2}} n^{D/2} + c_0 n^{D/2-1} \quad (\text{D2})$$

where $c_0 \sim \mathcal{O}(1)$ is the residue. Thus, assembling the expansions and plugging them into Equation (21) (to simplify notation we will write σ^2 instead of $\sigma^2(\{x_i\})$) we have

$$\begin{aligned} s(\{x_i\}, x_0) &\propto \left(\frac{n^{1/2}}{(2\pi)^{1/2}} + c_0 n^{-1/2} \right) \frac{(1 + n\sigma^2)^{(n+k-1)/2}}{(1 + n\sigma^2 + \Delta + r_0/n)^{(n+k)/2}} \\ &= \left(\frac{(2\pi)^{-1/2} n^{1/2} + c_0 n^{-1/2}}{(n\sigma^2 + \Delta + 1 + r_0/n)^{1/2}} \right) \frac{(1 + n\sigma^2)^{(n+k-1)/2}}{(1 + n\sigma^2 + \Delta + r_0/n)^{(n+k-1)/2}} \\ &= \left(\frac{(2\pi)^{-1/2} n^{1/2} + c_0 n^{-1/2}}{(n\sigma^2 + \Delta + 1 + r_0/n)^{1/2}} \right) \frac{1}{(1 + (\Delta + r_0/n)/(1 + n\sigma^2))^{(n+k-1)/2}} \end{aligned}$$

where we set $D = 1$ and used the fact that in one dimension the determinant satisfies $|A| = A$ and $I = 1$. We are now ready to take the limit $n \rightarrow \infty$ as both factors in the formula for $s(\{x_i\})$ have finite limits. Indeed, for the first factor we have

$$\left(\frac{(2\pi)^{-1/2} n^{1/2} + c_0 n^{-1/2}}{(n\sigma^2 + \Delta + 1 + r_0/n)^{1/2}} \right) \rightarrow \frac{1}{(2\pi\sigma^2)^{1/2}} \quad (\text{D3})$$

As for the second factor, observe that

$$\left[1 + \frac{\Delta + r_0 n^{-1}}{1 + n\sigma^2} \right]^{(n+k-1)/2} = \left[1 + \frac{\Delta + r_0 n^{-1}}{1 + n\sigma^2} \right]^{(k-1)/2} \left[1 + \frac{\Delta + r_0 n^{-1}}{1 + n\sigma^2} \right]^{n/2} \quad (\text{D4})$$

The first term approaches unity when $n \rightarrow \infty$, as for the second term notice that the power also depends on n , and so it requires more careful treatment. First, expanding in $1/n$ we can write

$$1 + \frac{\Delta + r_0 n^{-1}}{1 + n\sigma^2} = 1 + \frac{\Delta}{n\sigma^2} + \frac{r'_0}{n^2} \quad (\text{D5})$$

where $r'_0 \sim \mathcal{O}(1)$ (i.e., the residue contains all correction of order $1/n^2$ or smaller). Then, taking the logarithm and using the standard Taylor expansion $\log(1+x) = x + \mathcal{O}(x^2)$ and

¹³ This follows from the expansion $\frac{\Gamma(x+\alpha)}{\Gamma(x+\beta)} = x^{\alpha-\beta}(1 + \mathcal{O}(x^{-1}))$ which holds for $\alpha, \beta \geq 0$ and $x \rightarrow \infty$ (Laforgia & Natalini, 2012).

1506 taking $n \rightarrow \infty$ we have

$$\log \left[1 + \frac{\Delta}{n\sigma^2} + \frac{r'_0}{n^2} \right]^{n/2} = \frac{n}{2} \left(\frac{\Delta}{n\sigma^2} + \frac{r'_0}{n^2} + \mathcal{O}(n^{-2}) \right) \rightarrow \frac{\Delta}{2\sigma^2} \quad (\text{D6})$$

1507 Note that this is essentially a manifestation of the following limit identity: if a_n is a
 1508 sequence that converges to a in the limit $n \rightarrow \infty$, then the sequence $(1 + \frac{a_n}{n})^n$ converges to
 1509 e^a (Lemma 2.3.14 in Casella and Berger, 2024). Combined with the pervious analysis this
 1510 also implies that any terms that decay faster than $1/n$ within the power brackets $(\cdot)^n$ yield
 1511 vanishing contributions and can be discarded. In any case, combining the results, we
 1512 conclude that

$$s(\{x_i\}, x_0) \rightarrow \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left(-\frac{(x_0 - \bar{x}(\{x_i\}))^2}{2\sigma^2(\{x_i\})} \right) \quad (\text{D7})$$

1513 which is the desired formula.

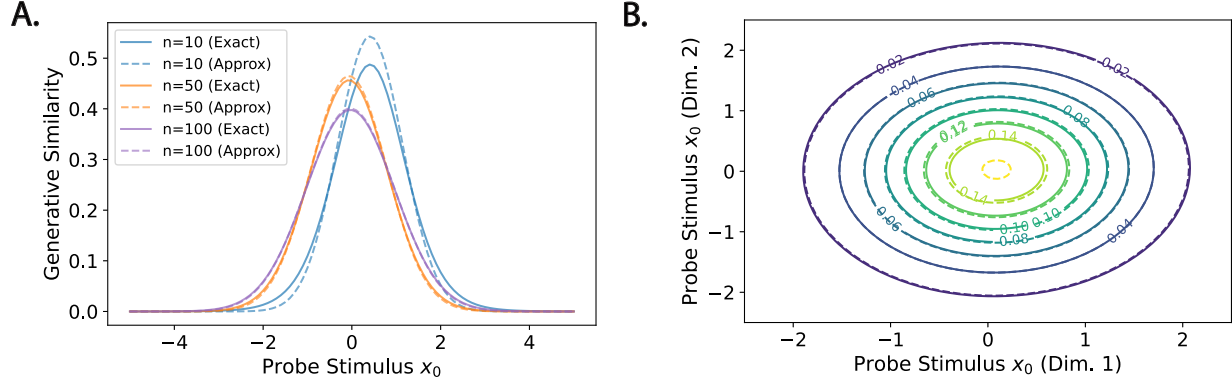
1514 The extension to $D > 1$ follows the same logic with some additional matrix
 1515 identities. First, recall that the scatter matrix can be written in terms of the empirical
 1516 covariance matrix $S(\{x_i\}) = n\Sigma(\{x_i\})$, and we assume that Σ remains finite and invertible
 1517 as $n \rightarrow \infty$ (otherwise a Gaussian generative model would be unsuitable). Second, we can
 1518 use the same expansion for the mean and covariance of the set $\{x_i\} \cup \{x_0\}$ and write

$$S(\{x_i\} \cup \{x_0\}) = n\Sigma(\{x_i\}) + \Delta + \mathcal{O}(n^{-1}) \quad (\text{D8})$$

1519 where $\Delta = (x_0 - \bar{x}(\{x_i\}))(x_0 - \bar{x}(\{x_i\}))^\top$. From here, we have

$$\begin{aligned} s(\{x_i\}, x_0) &\propto \frac{\mathcal{C}_{k,D,n+1}}{\mathcal{C}_{k,D,n}} \frac{|S(\{x_i\}) + I|^{(n+k-1)/2}}{|S(\{x_i\} \cup \{x_0\}) + I|^{(n+k)/2}} \\ &= \frac{\mathcal{C}_{k,D,n+1}}{\mathcal{C}_{k,D,n}} \frac{1}{|S(\{x_i\} \cup \{x_0\}) + I|^{1/2}} \frac{1}{|(S(\{x_i\}) + I)^{-1}(S(\{x_i\} \cup \{x_0\}) + I)|^{(n+k-1)/2}} \\ &\approx \frac{n^{D/2}}{(2\pi)^{D/2}} \frac{1}{|n\Sigma|^{1/2}} \frac{1}{|(I + \frac{1}{n}\Sigma^{-1}\Delta)^n|^{1/2}} \end{aligned}$$

1520 where in the second equality we used the following determinant identities $|A^{-1}| = |A|^{-1}$,
 1521 $|A^n| = |A|^n$, and $|AB| = |A||B|$ (Petersen, Pedersen, et al., 2008), and in the third equality
 1522 we kept only the leading terms in n that contribute to the limit $n \rightarrow \infty$. Now, using the


Figure D1

*Simulations of the $n \rightarrow \infty$ limit. **A.** Exact (solid) vs. large n approximation (dashed) generative similarity for one dimensional stimuli x_0 and a stimulus set $\{x_i\}$ of varying sizes n and a hyperparameter $k = 2$. The stimuli $\{x_i\}$ were sampled from a Gaussian distribution with zero mean and unit variance. **B.** Contour plot of exact (solid) and approximate (dashed) generative similarity for $D = 2$ dimensional stimuli and a stimulus set $\{x_i\}$ sampled from a Gaussian of zero mean and identity covariance matrix. We fixed a set size of $n = 100$ and a hyperparameter $k = 2$.*

1523 identities $|\alpha A| = \alpha^D |A|$ for any matrix A of dimensions $D \times D$ and scalar α , in addition to
 1524 the matrix exponential limit $\left(I + \frac{1}{n}A\right)^n \rightarrow e^A$ and the identity $|e^A| = e^{\text{tr}(A)}$ (Hall, 2013;
 1525 Petersen, Pedersen, et al., 2008), we arrive at

$$s(\{x_i\}, x_0) \rightarrow \frac{1}{\sqrt{(2\pi)^D |\Sigma|}} \exp\left(-\frac{1}{2} \text{tr}(\Sigma^{-1} \Delta)\right) \quad (\text{D9})$$

1526 Finally, observe that this is simply a different form of writing a multivariate Gaussian
 1527 distribution with empirical mean $\bar{x}(\{x_i\})$ and covariance matrix $\Sigma(\{x_i\})$. Indeed,

$$\begin{aligned} \text{tr}(\Sigma^{-1} \Delta) &= \text{tr}(\Sigma^{-1} (x_0 - \bar{x})(x_0 - \bar{x})^\top) \\ &= \sum_{i,j} (\Sigma^{-1})_{ij} (x_{0j} - \bar{x}_j)(x_{0i} - \bar{x}_i) \\ &= \sum_{i,j} (x_{0i} - \bar{x}_i) (\Sigma^{-1})_{ij} (x_{0j} - \bar{x}_j) \\ &= (x_0 - \bar{x})^\top \Sigma^{-1} (x_0 - \bar{x}) \end{aligned}$$

1528 From here, we conclude that

$$s(\{x_i\}, x_0) \rightarrow \frac{1}{\sqrt{(2\pi)^D |\Sigma(\{x_i\})|}} \exp \left(-\frac{1}{2} (x_0 - \bar{x}(\{x_i\}))^\top \Sigma^{-1}(\{x_i\}) (x_0 - \bar{x}(\{x_i\})) \right). \quad (\text{D10})$$

1529 This completes the derivation. As a sanity check, we confirmed in a simulation that the
 1530 theoretical limit and the exact formula indeed overlap as n grows larger (Figure D1).

Appendix E

Triplet Loss Formulation of Generative Similarity

1531 Given a generative model of the data, we can define a corresponding *contrastive* generative
1532 model on data triplets (X, X^+, X^-) as follows

$$p_c(x, x^+, x^-) = \int p(x|\theta^+)p(x^+|\theta^+)p(x^-|\theta^-)p(\theta^+)p(\theta^-)d\theta^+d\theta^- \quad (\text{E1})$$

1533 and given a triplet contrast function $\ell(x, x^+, x^-)$, e.g., $d(\phi(x), \phi(x^+)) - d(\phi(x), \phi(x^-))$ or
1534 some monotonic function of it (the dot product is also common; Sohn, 2016), we define the
1535 optimal embedding as

$$\varphi^* = \arg \min_{\varphi} \mathbb{E}_{p_c} \ell(\phi_{\varphi}(X), \phi_{\varphi}(X^+), \phi_{\varphi}(X^-)). \quad (\text{E2})$$

1536 Crucially, this function can be estimated via Monte Carlo sampling with triplets from

$$(X, X^+, X^-) \sim \begin{cases} \theta^+, \theta^- \sim p(\theta), & \text{Sample distributions} \\ x, x^+ \sim p(x|\theta^+), & \text{Sample 'same' examples} \\ x^- \sim p(x|\theta^-), & \text{Sample 'different' examples} \end{cases} \quad (\text{E3})$$

1537 Moreover, Equation (E2) has a desirable property: in Appendix F we prove that if ℓ is
1538 convex and strictly increasing in $\Delta_{\phi}(x, x^+, x^-) \equiv d(\phi(x), \phi(x^+)) - d(\phi(x), \phi(x^-))$ (e.g.,
1539 softmax loss $\ell(\Delta_{\phi}) = \log(1 + \exp(\Delta_{\phi}))$; Sohn, 2016), then the optimal encoder ϕ^* that
1540 minimizes (E2) ensures that the expected distance between same pairs is strictly smaller
1541 than that of different pairs as defined by the processes in Figure 1A, i.e.,

$$\mathbb{E}_{p_{\text{same}}} d(\phi^*(X), \phi^*(X^+)) < \mathbb{E}_{p_{\text{diff}}} d(\phi^*(X), \phi^*(X^-)). \quad (\text{E4})$$

Appendix F

Separation in Expectation

We want to show that for any triplet loss function $\ell(\Delta_\phi)$ that is convex and strictly increasing in $\Delta_\phi(x, x^+, x^-) = d_\phi(x, x^+) - d_\phi(x, x^-)$ where $d_\phi(x, y) = d(\phi(x), \phi(y))$ is a given embedding distance measure (e.g., softmax loss or quadratic loss), then the optimal embedding that minimizes (E2) ensures that the expected distance between same pairs is strictly smaller than that of different pairs as defined by the generative process in Figure 1A. To see this, let ϕ^* denote the optimal embedding and let ℓ^* denote its achieved loss. By definition, for any suboptimal embedding ϕ_{sub} which achieves ℓ_{sub} we have $\ell^* < \ell_{\text{sub}}$. One such suboptimal embedding (assuming non-degenerate distributions) is the constant embedding which collapses all samples into a point $\phi_{\text{sub}} = \phi_0$. In that case, we have $\Delta = 0$ and hence $\ell^* < \ell(0)$. Now, using Jensen inequality we have

$$\ell(0) > \ell^* = \mathbb{E}_{p_c} \ell(\Delta_{\phi^*}(X, X^+, X^-)) \geq \ell(\mathbb{E}_{p_c} \Delta_{\phi^*}(X, X^+, X^-)). \quad (\text{F1})$$

Observe next that since ℓ is strictly increasing (and hence its inverse is well-defined and strictly increasing) it follows that $\mathbb{E}_{p_c} \Delta_{\phi^*}(X, X^+, X^-) < 0$. Finally, by noting that

$$\begin{aligned} \mathbb{E}_{p_c} d_{\phi^*}(X, X^+) &= \int p(x|\theta^+)p(x^+|\theta^+)p(x^-|\theta^-)p(\theta^+)p(\theta^-)d_{\phi^*}(x, x^+)dxdx^+dx^-d\theta^+d\theta^- \\ &= \int p(x|\theta^+)p(x^+|\theta^+)p(\theta^+)d_{\phi^*}(x, x^+)dxdx^+d\theta^+ \\ &= \mathbb{E}_{p_{\text{same}}} d_{\phi^*}(X, X^+) \end{aligned}$$

and likewise,

$$\begin{aligned} \mathbb{E}_{p_c} d_{\phi^*}(X, X^-) &= \int p(x|\theta^+)p(x^+|\theta^+)p(x^-|\theta^-)p(\theta^+)p(\theta^-)d_{\phi^*}(x, x^-)dxdx^+dx^-d\theta^+d\theta^- \\ &= \int p(x|\theta^+)p(x^-|\theta^-)p(\theta^+)p(\theta^-)d_{\phi^*}(x, x^-)dxdx^-d\theta^+d\theta^- \\ &= \mathbb{E}_{p_{\text{diff}}} d_{\phi^*}(X, X^-) \end{aligned}$$

we arrive at the desired result

$$\mathbb{E}_{p_{\text{same}}} d_{\phi^*}(X, X^+) < \mathbb{E}_{p_{\text{diff}}} d_{\phi^*}(X, X^-). \quad (\text{F2})$$

Appendix G

Illustrative Example: Generative Similarity of a Gaussian Mixture

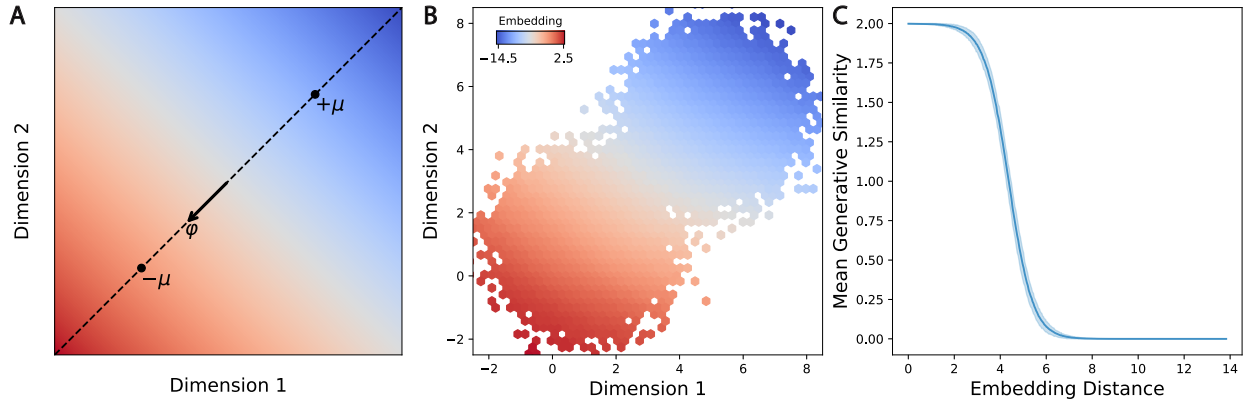


Figure G1

*Encoding the generative similarity of a Gaussian mixture. **A.** Optimal linear projection vector φ for a symmetric Gaussian mixture with means $\pm\mu$. **B.** Learned 1D embedding values from a two-layer perceptron for points sampled from a 2D Gaussian mixture (colors indicate values). **C.** Mean generative similarity as a function of distance in the embedding space shown in **B** (discretized into 500 quantile bins). Shaded area indicates 95% CIs bootstrapped over data points.*

To get a sense of the contrastive training procedure with generative similarity, we consider a Gaussian mixture example. Gaussian mixtures are ideal because they are analytically and numerically tractable. Consider a data generative process that is given by a mixture of two Gaussians with means $\mu_{1,2}$, equal variances σ^2 , and a uniform prior $p_{1,2} = 1/2$. Without loss of generality, we can choose a coordinate system in which $\mu_1 = -\mu_2 \equiv \mu$. Consider further a subfamily of embeddings that are specified by linear projections $\phi_\varphi(x) = \varphi \cdot x$ where φ is a unit vector of choice $\varphi \cdot \varphi = 1$. Given the contrastive sampling process p_c of Equation (E1), a natural measure of contrastive loss in this case would be

$$\mathbb{E}_{p_c} \ell = \mathbb{E}_{p_c} [\phi_\varphi(X) \cdot \phi_\varphi(X^-) - \phi_\varphi(X) \cdot \phi_\varphi(X^+)] \quad (\text{G1})$$

i.e., using the ‘dot’ product as a measure of embedding similarity (Sohn, 2016) (for

one-dimensional embeddings this is just a regular product). By plugging in the definition of the Gaussian mixture and simplifying using Gaussian moments (see below), Equation (G1) boils down to $\mathbb{E}_{p_c} \ell \propto -4\|\mu\|_2^2 \cos^2 \theta_{\varphi\mu}$ where $\theta_{\varphi\mu}$ is the angle between φ and μ . This loss is minimized for $\cos \theta_{\varphi\mu} = \pm 1$ or equivalently $\varphi^* = \pm \hat{\mu}$ where $\hat{\mu}$ is the normalized version of μ . This means that the optimal linear mapping is simply one that projects different points onto the axis connecting the centers of the two Gaussians $\mu_1 - \mu_2 = 2\mu$, which is equivalent to a linear decision boundary that is orthogonal to the line $\mu_1 - \mu_2$ and passing through the origin, and thus effectively recovering a linear classifier (Figure G1A). We further confirmed in a simulation that this behavior persists when applying the Monte-Carlo approximation in Equation (E3) with a quadratic loss to a two-layer perceptron (Figure G1B; classification test accuracy of 99.7%; additional details below). Moreover, we compared distance in the learned embedding space to the theoretical generative similarity values in that case and found excellent agreement (Spearman's $\rho(498) = -0.99$, $p < 10^{-3}$; Figure G1C). We also found that the 95% confidence interval (CI) on the average distance for same pairs was $[1.62, 1.66]$ whereas for different pairs it was $[4.77, 4.91]$, consistent with the prediction of Equation (E4).

Analytical Derivation

To derive the linear projection result, we start by plugging in the definition of the Gaussian mixture generative process into Equation (E2)

$$\begin{aligned}
 \mathbb{E}_{p_c} \ell(X, X^+, X^-) &= \sum_{i=1,2} \sum_{j=1,2} \int \ell(\phi(x), \phi(x^+), \phi(x^-)) \times \\
 &\quad \times \frac{1}{4} \frac{1}{((2\pi)^d \sigma^{2d})^{3/2}} \exp\left(-\frac{(x - \mu_i)^2 + (x^+ - \mu_i)^2 + (x^- - \mu_j)^2}{2\sigma^2}\right) dx dx^+ dx^-
 \end{aligned}$$

Now, recall that

$$\begin{aligned}
 \ell(x, x^+, x^-) &= \phi_\varphi(x) \cdot \phi_\varphi(x^-) - \phi_\varphi(x) \cdot \phi_\varphi(x^+) \\
 &= (\varphi \cdot x)(\varphi \cdot x^-) - (\varphi \cdot x)(\varphi \cdot x^+)
 \end{aligned}$$

1586 Substituting into the loss formula and integrating we have

$$\begin{aligned}\mathbb{E}_{p_c} \ell &\propto \sum_{i,j} \int (\varphi \cdot x)(\varphi \cdot x^-) \exp \left(-\frac{(x - \mu_i)^2 + (x^- - \mu_j)^2}{2\sigma^2} \right) dx dx^- \\ &\quad - 2 \sum_i \int (\varphi \cdot x)(\varphi \cdot x^+) \exp \left(-\frac{(x - \mu_i)^2 + (x^+ - \mu_i)^2}{2\sigma^2} \right) dx dx^+\end{aligned}$$

1587 Note next that since x^+ and x^- are dummy integration variables, we can further rewrite

$$\begin{aligned}\mathbb{E}_{p_c} \ell &\propto \sum_{i \neq j} \int (\varphi \cdot x)(\varphi \cdot x^-) \exp \left(-\frac{(x - \mu_i)^2 + (x^- - \mu_j)^2}{2\sigma^2} \right) dx dx^- \\ &\quad - \sum_i \int (\varphi \cdot x)(\varphi \cdot x^+) \exp \left(-\frac{(x - \mu_i)^2 + (x^+ - \mu_i)^2}{2\sigma^2} \right) dx dx^+\end{aligned}$$

1588 Thus, using the fact that the distributions are separable and standard Gaussian moment
1589 formulae we arrive at

$$\mathbb{E}_{p_c} \ell \propto \sum_{i \neq j} (\varphi \cdot \mu_i)(\varphi \cdot \mu_j) - \sum_i (\varphi \cdot \mu_i)(\varphi \cdot \mu_i) \quad (\text{G2})$$

1590 Finally, plugging in $\mu_1 = -\mu_2 = \mu$ and using the fact that $\|\varphi\|_2^2 = \varphi \cdot \varphi = 1$ we have

$$\mathbb{E}_{p_c} \ell \propto -4(\varphi \cdot \mu)^2 = -4\|\mu\|_2^2 \cos^2 \theta_{\varphi\mu}. \quad (\text{G3})$$

1591 Simulation Details

1592 To test the Monte Carlo approximation of Equation (E3) and to see how well it
1593 tracks the theoretical generative similarity, we considered an embedding family that is
1594 parametrized by two-layer perceptrons. For the generative family, we chose as before a
1595 mixture of two Gaussians, this time with mean values of $\mu_1 = (5, 5)$ and $\mu_2 = (1, 1)$ and
1596 unit variance $\sigma^2 = 1$. As for the loss function, here we used a quadratic (Euclidean) loss of
1597 the form

$$\mathcal{L} = \frac{1}{N_{\text{triplets}}} \sum_{\{x, x^+, x^-\}} [(\phi(x) - \phi(x^+))^2 - (\phi(x) - \phi(x^-))^2] \quad (\text{G4})$$

1598 where $\{x, x^+, x^-\}$ are triplets sampled from Equation (E3). We trained the perceptron
1599 model using 10,000 triplets (learning rate = 10^{-5} , hidden layer size = 32, batch-size = 256,
1600 and 300 epochs). The resulting model successfully learned to distinguish the two Gaussians

as seen visually from the embedding values in Figure G1B, and also from the test accuracy of 99.7%. Finally, we wanted to see how well the embedding distance tracked the theoretical generative similarity, which in this case can be derived in closed form by simply plugging in the Gaussian distributions in Equation (6)

$$s(x_1, x_2) = \frac{\frac{1}{2}e^{-\frac{(x_1-\mu_1)^2}{2\sigma^2}} \times e^{-\frac{(x_2-\mu_1)^2}{2\sigma^2}} + \frac{1}{2}e^{-\frac{(x_1-\mu_2)^2}{2\sigma^2}} \times e^{-\frac{(x_2-\mu_2)^2}{2\sigma^2}}}{\left(\frac{1}{2}e^{-\frac{(x_1-\mu_1)^2}{2\sigma^2}} + \frac{1}{2}e^{-\frac{(x_1-\mu_2)^2}{2\sigma^2}}\right) \times \left(\frac{1}{2}e^{-\frac{(x_2-\mu_1)^2}{2\sigma^2}} + \frac{1}{2}e^{-\frac{(x_2-\mu_2)^2}{2\sigma^2}}\right)}. \quad (\text{G5})$$

The mean generative similarity as a function of embedding distance between pairs (grouped into 500 quantile bins) is shown in Figure G1C and exhibiting high consistency.

Appendix H

Generative Similarity for Quadrilateral Shapes

1607 The hard feature-class limit corresponds to taking $\alpha = \beta \rightarrow 0$ in Equation C8. In this case,
 1608 the Beta distributions that generate Bernoulli parameters become highly concentrated
 1609 around 0 and 1 since $\text{Beta}(\theta_i; \alpha, \beta) \propto \theta_i^{\alpha-1}(1 - \theta_i)^{\beta-1}$. From here, we have

$$\log s(\sigma_1(f^{(1)}), \sigma_2(f^{(2)})) = n \log 2 - \log \frac{\beta + 1}{\beta} (|\Sigma_1 - \Sigma_2| + |\Sigma_2 - \Sigma_1|) \quad (\text{H1})$$

1610 where the first term is simply a constant. Next, observe that

$$\begin{aligned}
 |\Sigma_1 - \Sigma_2| + |\Sigma_2 - \Sigma_1| &= \sum_i f_i^{(1)}(1 - f_i^{(2)}) + f_i^{(2)}(1 - f_i^{(1)}) \\
 &= \sum_i f_i^{(1)} - 2f_i^{(1)}f_i^{(2)} + f_i^{(2)} \\
 &= \sum_i (f_i^{(1)} - f_i^{(2)})^2
 \end{aligned} \quad (\text{H2})$$

1611 where the third equality follows from the fact that $f^2 = f$ for binary features. Thus,
 1612 generative similarity reduces to a monotonically decreasing function of the Euclidean
 1613 distance between the geometric features of shapes

$$\log s(\sigma_1(f^{(1)}), \sigma_2(f^{(2)})) = n \log 2 - \log \frac{\beta + 1}{\beta} \sum_i (f_i^{(1)} - f_i^{(2)})^2 \quad (\text{H3})$$

1614 which is the desired result.

Appendix I

Details on Quadrilateral Experiment

For our main experiments, we use the CorNet model which was used in the original work (variant ‘S’) that introduces the Oddball task (Sablé-Meyer et al., 2021). CorNet contains four “areas” corresponding to the areas of the visual stream: V1, V2, V4, and IT. Each area contains convolutional and max pooling layers. There are also biologically plausible recurrent connections between areas (e.g., V4 to V1). After IT, the penultimate area in the visual stream, a linear layer is used to readout object categories. The model is pretrained on ImageNet-1k on a standard supervised image classification objective.

For finetuning the model on the generative similarity contrastive objective, we first calculated the Euclidean distance of the model’s final layer embedding between different quadrilateral images, then calculated the Euclidean distance between the quadrilaterals’ respective geometry feature vectors, and finally used the mean squared error between the embeddings’ distance and the feature vectors’ distance as the loss. The geometric feature vectors were a set of 22 binary features encoding the following properties: 6 features per pair of edges encoding whether their lengths are equal or not, 6 features per pair of angles coding whether their angles are equal or not, 6 features per pair of edges encoding whether they were parallel or not, and 4 features per angle encoding whether they were right angles or not. See Table I1 for a list of these values. For training data, we used quadrilaterals from all 11 categories with different scales and rotations (though the specific quadrilateral images used in the test trials were held out). We used the Adam optimizer with a learning rate of $5e-4$. We conducted 10 training runs.

We used the exact same training data, learning rate, and optimizer when running the control experiments for finetuning CorNet on the SimCLR objective. For the SimCLR objective, the augmentations we used were random resize crop, random horizontal flips, and random Gaussian blurs. The original SimCLR paper also had augmentations related to color distortions that we did not use because our data were already grayscale images.

Like SimCLR, we used the InfoNCE loss function (Oord et al., 2018). Let v_i be the embedding of image i and v'_i be the embedding of image i 's augmented counterpart. The loss is $\frac{1}{N} \sum_{i=1}^N \log \frac{f(v_i, v'_i)}{\frac{1}{N} \sum_j \exp f(v_i, v'_j)}$ where f is a similarity function between embeddings (SimCLR used cosine similarity). This effectively pushes representations of images and their augmented counterparts to be more similar while also pushing representations of images and other images' augmented counterparts to be more dissimilar. Training was conducted on an NVIDIA Quadro P6000 GPU with 25GB of memory.

For finetuning the model on the supervised classification objective, we followed the protocol used in the supplementary results of Sablé-Meyer et al. (2021). Specifically, 11 new object categories are added to the model's last layer, and the model is trained to classify a quadrilateral as one of the 11 categories shown in Figure 6B. Like the contrastive settings, we used training data from each category of quadrilaterals with different scales and rotations (specific images used in the test trials were held out). We used a learning rate of 5e-6 using the Adam optimizer with a cross entropy loss. We conducted 10 training runs. Training was conducted on an NVIDIA Quadro P6000 GPU with 25GB of memory.

Table I1

List of geometric regularities for each quadrilateral type (sorted from most regular to least)

shape	rightAngles	parallels	symmetry	equalSides	equalAngles
square	4	2	4	4	4
rectangle	4	2	2	2	4
losange	0	0	2	4	2
parallelogram	0	2	1	2	2
rightKite	2	0	1	2	2
kite	0	0	1	2	2
isoTrapezoid	0	1	1	1	2
hinge	1	0	0	1	0
rustedHinge	0	0	0	1	0
trapezoid	0	1	0	0	0
random	0	0	0	0	0

Appendix J

Geoclidian Concepts

Below are the 37 programs presented in the original Geoclidian benchmark (Hsu et al., 2022). We used 30 for training and randomly held out 7 for testing ('cccc', 'clcc', 'quadrilateral', 'triangle', 'perp bisector', 'llcc', 'ccc'). Test concepts have four asterisks next to their names.

```

1659     perp_bisector****
1660     'l1 = line(p1(), p2())',
1661     'c1* = circle(p1(), p2())',
1662     'c2* = circle(p2(), p1())',
1663     'l2 = line(p3(c1, c2), p4(c1, c2))',
1664
1665     rectilinear
1666     'l1 = line(p1(), p2())',
1667     'l2 = line(p2(), p3())',
1668     'l3 = line(p3(), p4())',
1669     'l4 = line(p4(), p5())',
1670     'l5 = line(p5(), p1())',
1671
1672     ang_bisector
1673     'l1 = line(p1(), p2())',
1674     'c1* = circle(p1(), p2())',
1675     'c2* = circle(p2(), p1())',
1676     'c3* = circle(p3(c1, c2), p1())',
1677     'l2 = line(p1(), p3(c1, c2))',
1678     'l3 = line(p1(), p4(c2, c3))',
1679

```

```

1680      oblong
1681      'l1* = line(p1(), p2())',
1682      'c1* = circle(p1(), p2())',
1683      'c2* = circle(p2(), p1())',
1684      'l2* = line(p3(c1, c2), p4(c1, c2))',
1685      'c3* = circle(p5(l2), p6(l2))',
1686      'c4* = circle(p6(l2), p5(l2))',
1687      'l3* = line(p7(c3, c4), p8(c3, c4))',
1688      'c5* = circle(p9(l3), p10(l3))',
1689      'c6* = circle(p10(l3), p9(l3))',
1690      'l4* = line(p11(c5, c6), p12(c5, c6))',
1691      'l5 = line(p13(l1, l2), p14(l2, l3))',
1692      'l6 = line(p14(l2, l3), p15(l3, l4))',
1693      'l7 = line(p15(l3, l4), p16(l1, l4))',
1694      'l8 = line(p16(l1, l4), p13(l1, l2))',
1695
1696      eq_t
1697      'l1 = line(p1(), p2())',
1698      'c1* = circle(p1(), p2())',
1699      'c2* = circle(p2(), p1())',
1700      'l2 = line(p1(), p3(c1, c2))',
1701      'l3 = line(p2(), p3())',
1702
1703      rhombus
1704      'l1* = line(p1(), p2())',
1705      'c1* = circle(p1(), p2())',
1706      'c2* = circle(p2(), p1())',

```

```

1707     'l2* = line(p3(c1, c2), p4(c1, c2))',
1708     'l3 = line(p1(), p3(c1, c2))',
1709     'l4 = line(p3(c1, c2), p2())',
1710     'l5 = line(p2(), p4(c1, c2))',
1711     'l6 = line(p4(c1, c2), p1())',
1712
1713     parallel_l
1714     'l1* = line(p1(), p2())',
1715     'c1* = circle(p1(), p3(l1))',
1716     'c2* = circle(p3(l1), p1())',
1717     'c3* = circle(p2(), p3(l1))',
1718     'l2 = line(p4(c1, c2), p5(c1, c2))',
1719     'l3 = line(p6(c2, c3), p7(c2, c3))',
1720
1721     rhomboid
1722     'l1* = line(p1(), p2())',
1723     'c1* = circle(p1(), p3(l1))',
1724     'c2* = circle(p3(l1), p1())',
1725     'c3* = circle(p4(l1, c2), p3(l1))',
1726     'c4* = circle(p5(l1, c3), p6(l1, c2))',
1727     'l2 = line(p1(), p7(c1, c2))',
1728     'l3 = line(p7(c1, c2), p8(c3, c4))',
1729     'l4 = line(p8(c3, c4), p4(l1, c2))',
1730     'l5 = line(p4(l1, c2), p1())',
1731
1732
1733

```

```

1734     angle
1735     'l1 = line(p1(), p2())',
1736     'l2 = line(p2(), p3())',
1737
1738     square
1739     'l1* = line(p1(), p2())',
1740     'c1* = circle(p1(), p2())',
1741     'c2* = circle(p2(), p1())',
1742     'l2* = line(p3(c1, c2), p4(c1, c2))',
1743     'c3* = circle(p5(l1, l2), p1())',
1744     'l3 = line(p1(), p6(l2, c3))',
1745     'l4 = line(p6(l2, c3), p2())',
1746     'l5 = line(p2(), p7(l2, c3))',
1747     'l6 = line(p7(l2, c3), p1())',
1748
1749     sixty_ang
1750     'l1 = line(p1(), p2())',
1751     'c1* = circle(p1(), p2())',
1752     'c2* = circle(p2(), p1())',
1753     'l2 = line(p1(), p3(c1, c2))',
1754
1755     right_ang_t
1756     'l1* = line(p1(), p2())',
1757     'c1* = circle(p1(), p2())',
1758     'c2* = circle(p2(), p1())',
1759     'l2* = line(p3(c1, c2), p4(c1, c2))',
1760     'l3 = line(p5(l1, l2), p6(l2))',

```

```

1761     'l4 = line(p5(l1, l2), p7(l1))',
1762     'l5 = line(p6(l2), p7(l1))',
1763
1764     radii
1765     'c1 = circle(p1(), p2())',
1766     'l1 = line(p1(), p3(c1))',
1767     'l2 = line(p1(), p4(c1))',
1768     'l3 = line(p1(), p5(c1))',
1769
1770     diameter
1771     'l1* = line(p1(), p2())',
1772     'c1* = circle(p1(), p2())',
1773     'c2* = circle(p2(), p1())',
1774     'l2 = line(p3(c1, c2), p4(c1, c2))',
1775     'c3 = circle(p5(l1, l2), p3(c1, c2))',
1776
1777     triangle****
1778     'l1 = line(p1(), p2())',
1779     'l2 = line(p2(), p3())',
1780     'l3 = line(p3(), p1())',
1781
1782     quadrilateral****
1783     'l1 = line(p1(), p2())',
1784     'l2 = line(p2(), p3())',
1785     'l3 = line(p3(), p4())',
1786     'l4 = line(p4(), p1())',
1787

```



```

1788     segment
1789     'c1 = circle(p1(), p2())',
1790     'l1 = line(p3(c1), p4(c1))',
1791
1792     llcc****
1793     'l1 = line(p1(), p2())',
1794     'l2 = line(p3(), p4())',
1795     'c3 = circle(p5(l1), p6(l1, l2))',
1796     'c4 = circle(p7(), p8(l1, l2))',
1797
1798     ccl
1799     'c1 = circle(p1(), p2())',
1800     'c2 = circle(p3(c1), p4())',
1801     'l3 = line(p5(c1), p6(c1, c2))',
1802
1803     ccc****
1804     'c1 = circle(p1(), p2())',
1805     'c2 = circle(p3(c1), p4(c1))',
1806     'c3 = circle(p5(c1), p6(c1, c2))',
1807
1808     clcl
1809     'c1 = circle(p1(), p2())',
1810     'l2 = line(p3(c1), p4(c1))',
1811     'c3 = circle(p5(), p6(c1, l2))',
1812     'l4 = line(p7(c1), p8(c3))',
1813
1814

```

```

1815      clt
1816      'c1 = circle(p1(), p2())',
1817      'l2 = line(p3(c1), p4())',
1818      'l3 = line(p5(c1, l2), p6(c1))',
1819      'l4 = line(p6(), p7(c1))',
1820      'l5 = line(p7(), p5())',
1821
1822      llt
1823      'l1 = line(p1(), p2())',
1824      'l2 = line(p3(), p4())',
1825      'l3 = line(p5(l1, l2), p6())',
1826      'l4 = line(p6(), p7())',
1827      'l5 = line(p7(), p5())',
1828
1829      clcc****
1830      'c1 = circle(p1(), p2())',
1831      'l2 = line(p3(c1), p4(c1))',
1832      'c3 = circle(p5(c1), p6(c1))',
1833      'c4 = circle(p7(c1), p8(c1))',
1834
1835      lllc
1836      'l1 = line(p1(), p2())',
1837      'l2 = line(p3(l1), p4())',
1838      'l3 = line(p5(l2), p6())',
1839      'c4 = circle(p7(l1, l2), p8(l3))',
1840
1841

```

```
1842      cccc****
1843      'c1 = circle(p1(), p2())',
1844      'c2 = circle(p3(c1), p4(c1))',
1845      'c3 = circle(p5(c1, c2), p6(c1, c2))',
1846      'c4 = circle(p7(c2, c3), p8(c2, c3))',
1847
1848      cccl
1849      'c1 = circle(p1(), p2())',
1850      'c2 = circle(p3(), p4(c1))',
1851      'c3 = circle(p5(), p6(c1, c2))',
1852      'l4 = line(p7(c1, c2), p8(c1, c2))',
1853
1854      clll
1855      'c1 = circle(p1(), p2())',
1856      'l2 = line(p3(c1), p4())',
1857      'l3 = line(p5(l2), p6())',
1858      'l4 = line(p7(c1, l2), p8(l3))',
1859
1860      llll
1861      'l1 = line(p1(), p2())',
1862      'l2 = line(p3(), p4())',
1863      'l3 = line(p5(l1, l2), p6())',
1864      'l4 = line(p7(l1, l2), p8())',
1865
1866      tll
1867      'l1 = line(p1(), p2())',
1868      'l2 = line(p1(), p3())',
```

```

1869     'l3 = line(p2(), p3())',
1870     'l4 = line(p4(l1, l2), p5())',
1871     'l5 = line(p6(l1, l2), p7())',
1872
1873     c1l
1874     'c1 = circle(p1(), p2())',
1875     'l2 = line(p3(c1), p4(c1))',
1876     'l3 = line(p5(c1), p6(c1))',
1877
1878     l1l
1879     'l1 = line(p1(), p2())',
1880     'l2 = line(p3(l1), p4())',
1881     'l3 = line(p5(l1), p6())',
1882
1883     llc
1884     'l1 = line(p1(), p2())',
1885     'l2 = line(p3(), p4(l1))',
1886     'c3 = circle(p5(l1), p6(l1, l2))',
1887
1888     cct
1889     'c1 = circle(p1(), p2())',
1890     'c2 = circle(p3(), p4())',
1891     'l3 = line(p5(c1, c2), p6(c1, c2))',
1892     'l4 = line(p6(), p7(c2))',
1893     'l5 = line(p7(), p5())',
1894
1895

```

```
1896      tcl
1897      'l1 = line(p1(), p2())',
1898      'l2 = line(p1(), p3())',
1899      'l3 = line(p2(), p3())',
1900      'c4 = circle(p4(l1, l2), p5())',
1901      'l5 = line(p6(l1, l2), p7(l3))',
1902
1903      lcc
1904      'l1 = line(p1(), p2())',
1905      'c2 = circle(p3(l1), p4(l1))',
1906      'c3 = circle(p5(l1, c2), p6(l1, c2))',
1907
1908      tcc
1909      'l1 = line(p1(), p2())',
1910      'l2 = line(p1(), p3())',
1911      'l3 = line(p2(), p3())',
1912      'c4 = circle(p4(l1, l2), p5())',
1913      'c5 = circle(p6(l1, l2), p7(l3))',
```

Appendix K

Geoclidean Context-Free Grammar

1914 Context free grammar used to parse Geoclidean programs into parse trees.

1915 `S -> ConstructionSequence`

1916

1917 `ConstructionSequence -> Construction | Construction ConstructionSequence`

1918

1919 `Construction -> ObjectDef '=' Object`

1920

1921 `ObjectDef -> VisibleObject | InvisibleObject`

1922 `VisibleObject -> OBJECT_NAME`

1923 `InvisibleObject -> OBJECT_NAME '*'`

1924

1925 `Object -> Line | Circle`

1926

1927 `Line -> 'line' '(' PointRef ',' PointRef ')'`

1928 `Circle -> 'circle' '(' PointRef ',' PointRef ')'`

1929

1930 `PointRef -> SimplePoint | ConstrainedPoint`

1931 `SimplePoint -> Point EmptyArgs`

1932 `ConstrainedPoint -> Point ConstraintArgs`

1933

1934 `Point -> POINT_NAME`

1935

1936 `EmptyArgs -> '(' ')'`

1937 `ConstraintArgs -> '(' Constraints ')'`

1938

```
1939      Constraints -> SingleConstraint | DoubleConstraint
1940
1941      SingleConstraint -> ObjectRef
1942
1943      DoubleConstraint -> ObjectRef ', ' ObjectRef
1944
1945      ObjectRef -> OBJECT_NAME | OBJECT_NAME '*'
1946
1947      OBJECT_NAME -> LINE_NAME | CIRCLE_NAME
1948
1949      LINE_NAME -> 'l1' | 'l2' | 'l3' | 'l4' | 'l5' | 'l6' | 'l7' | 'l8' | 'l9'
1950
1951      CIRCLE_NAME -> 'c1' | 'c2' | 'c3' | 'c4' | 'c5' | 'c6'
1952
1953
1954      POINT_NAME -> 'p1' | 'p2' | 'p3' | 'p4' | 'p5' | 'p6' | 'p7' | 'p8' ... | 'p16'
```

Appendix L

Details on Geoclidean Experiment

Out of the 37 programs or “concepts” presented in the original Geoclidean work (shown in Appendix J), we randomly chose 30 to generate our training data and the other 7 for testing. We generated a dataset of 100,000 images from the training concepts. For all of our models, we finetuned the pretrained ResNet-50 architecture (He et al., 2016) that was pretrained on ImageNet. We conducted 4 different training runs for each condition. Each model was trained for 35 epochs using the Adam optimizer with a batch size of 128. Images were resized to $224 \times 224 \times 3$ (for 3 RGB channels) before training.

For the SimCLR baseline objective, images in the batch were randomly augmented. The augmentations were: random resize crop, random horizontal flips, and random Gaussian blurs. The original SimCLR paper also had augmentations corresponding to color distortions which we did not use because our data were already grayscale images. Like SimCLR, we used the InfoNCE loss function (Oord et al., 2018). Let v_i be the embedding of image i and v'_i be the embedding of image i ’s augmented counterpart. The loss is $\frac{1}{N} \sum_{i=1}^N \log \frac{f(v_i, v'_i)}{\frac{1}{N} \sum_j \exp f(v_i, v'_j)}$ where f is a similarity function between embeddings (SimCLR used cosine similarity). This effectively pushes representations of images and their augmented counterparts to be more similar while also pushing representations of images and other images’ augmented counterparts to be more dissimilar. Training was conducted with one NVIDIA Tesla P100 GPU with 16GB of memory.

For the generative similarity objective, let image a be the the representation of an image that is sampled from a random program c and image b be another image sampled from a program d (which may or may not be the same program). We compute the mean squared error between the cosine similarity between a and b and the logarithm of the tree kernel between the respective parse trees of c and d . The tree kernel from Moschitti (2006) recursively counts number of shared subtrees. Although Moschitti (2006) also propose a method for subset trees, which is more exhaustive than parse trees, we found our results

are sufficiently obtained through counting shared subtrees.

For the Monte-Carlo (class-based) baseline objective, let image a be the representation of the anchor image that is sampled from a random program c . Let p be the representation of a positive image that is another image sampled from c and n be the negative image that is randomly sampled from another program. Using these samples, we compute the positive Euclidean distance $d(p, a)$ and the negative Euclidean distance $d(n, a)$ in order to compute the loss function $d(p, a) - d(n, a)$. This loss is equivalent to an implementation of supervised contrastive learning (Khosla et al., 2020). Note that since the triplet loss is defined with respect to Euclidean distance in this case, we used this measure of distance instead of cosine similarity when making predictions based on embeddings.

Each Geoclidan problem consists of a small set of reference images generated from a specific geometric program, along with a set of test images. Some test images are derived from the same underlying program (positives), while others are distractors generated from different programs. Distractor difficulty varies across two problem types: far problems involve distractors that differ substantially from the reference program, while close problems include distractors with only minimal changes, making classification more challenging.

To evaluate performance, we extract image embeddings using a CNN and compute the mean embedding of the reference set to define a prototype representation. A test image is classified as in-concept if its embedding lies within a certain threshold from this prototype. The threshold is chosen to maximize classification performance across tasks following Hsu et al. (2022).

Appendix M

Generative Similarity of Items Sampled from a Hierarchy

We want to compute the generative similarity between a pair of items sampled from a hierarchy (similar to Figure 9A). Specifically, we assume that items are sampled from a tree hierarchy by probabilistically choosing a sequence of branches from the root until a leaf is reached, after which an item is selected uniformly from a set of items associated with that leaf. We assume that each possible item is associated with a single leaf, such that there is a unique path from the root to the relevant leaf. Given two items x_i and x_j , let K denote the branching or separation depth at which their trajectories from the root to their respective leaves diverge. Moreover, let N_i and N_j denote the depth of each respective leaf, and let M_i and M_j denote the size of the set of items associated with each leaf. From here, under the ‘different’ (i.e., independent) sampling assumption we can denote the probabilities of sampling x_i and x_j as $p_1 p_2 \dots p_K p_{K+1}^{(i)} \dots p_{N_i}^{(i)} \times \frac{1}{M_i}$ and $p_1 p_2 \dots p_K p_{K+1}^{(j)} \dots p_{N_j}^{(j)} \times \frac{1}{M_j}$ for x_i and x_j , respectively, where each p is the probability of (uniformly) sampling a branch at each branching point. Note that up until the branching point the probabilities are shared by definition. On the other hand, when x_i and x_j are sampled jointly, any joint sampling process that traverses from the root along the shared path and stops no later than the branching point is a valid joint sampling process for x_i and x_j and so we must sum over all of them. Thus, overall we have

$$\begin{aligned}
 s(x_i, x_j) &= \frac{1}{K} \frac{1}{M_i M_j} \frac{\left[p_1 (p_2 \dots p_K \times p_{K+1}^{(i)} \dots p_{N_i}^{(i)}) (p_2 \dots p_K \times p_{K+1}^{(j)} \dots p_{N_j}^{(j)}) + \dots \right]}{p_1 p_2 \dots p_K p_{K+1}^{(i)} \dots p_{N_i}^{(i)} \times \frac{1}{M_i} \times p_1 p_2 \dots p_K p_{K+1}^{(j)} \dots p_{N_j}^{(j)} \times \frac{1}{M_j}} \\
 &= \frac{1}{K} \frac{p_1 (p_2 \dots p_K)^2 + p_1 p_2 (p_3 \dots p_K)^2 + \dots + p_1 p_2 \dots p_K}{(p_1 p_2 \dots p_K)^2} \\
 &= \frac{1}{K} \left(\frac{1}{p_1} + \frac{1}{p_1 p_2} + \dots + \frac{1}{p_1 p_2 \dots p_K} \right)
 \end{aligned}$$

where in the second and third equalities we simply eliminated shared factors that cancel out. This recovers Equation (35) which is the desired result.

Appendix N

Details on Natural Image Hierarchy Experiment

Our training data came from the standard ImageNet-1k dataset (ILSVRC 2012) (Deng et al., 2009). This dataset contains over a million training images from 1000 different categories. These categories are each considered leaf nodes inside the WordNet tree hierarchy and are non-overlapping. Images were resized to $224 \times 224 \times 3$ (for 3 RGB channels) before training. We trained a standard ResNet-50 (He et al., 2016) architecture on all types of losses. We conducted 5 different training runs for each condition. For all objectives, we trained for 65 epochs, we used the same learning rate $1e-3$ and the same Adam optimizer with a batch size of 256. Training was conducted with one NVIDIA A100 GPU with 40GB of memory.

For the SimCLR baseline objective, we used the same objective and image augmentations as the previous experiments, but also included random grayscaling since ImageNet images are colored. For the class-based baseline objective, we used a similar triplet-based loss as in the Geoclidean experiment, with an anchor and positive image from the same category and a negative example from a different category (correspondingly, we used Euclidean distance to train and derive similarity predictions). For the generative similarity objective, we computed the mean-squared error between the cosine similarities between two image embeddings and their log generative similarity measure as defined in Equation 35 (we used the logarithm as it provided more stable training).

Appendix O

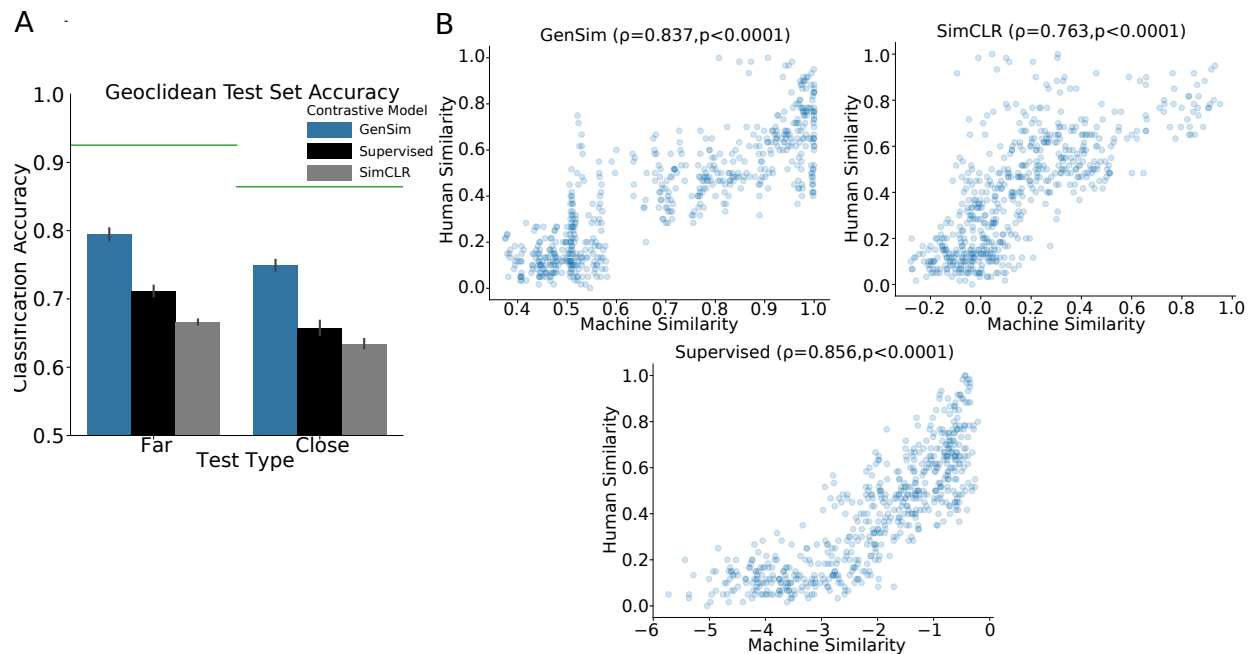
Experiments with Supervised Contrastive Learning

In the main text of this work, we use SimCLR as a baseline to our contrastive generative similarity approach. As a reminder, SimCLR is a training framework whereby a representation is learned by applying various augmentations to the training data and then using those to construct positive and negative pairs that are then incorporated in a contrastive (e.g. triplet) loss. Crucially, positive and negative pairs in SimCLR (and InfoNCE; Oord et al., 2018) are usually constructed in a domain-agnostic manner by applying augmentations such as rotations and rescalings to the data. While this is a good default strategy, when there is prior expectation about the structure of the underlying generative process, it glosses over finer structure present in the data, such as within- and across-category relations. Supervised contrastive learning (SupCon, Khosla et al., 2020) goes one step in this direction by defining same-different relations with respect to meaningful semantic labels.

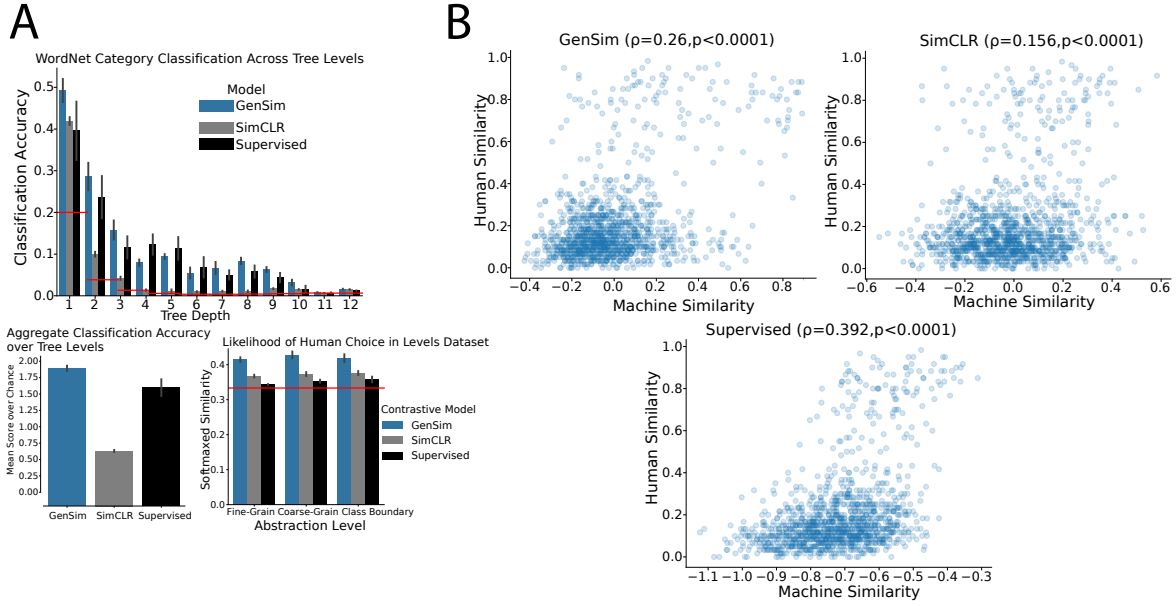
Note that supervised contrastive learning can be captured in the generative similarity framework by assuming independent, finite, and discrete distributions for every class. However, our generative similarity approach expands the expressivity of contrastive learning by moving from similarity over labels to similarity over distributions, which can capture more nuanced, complex, and graded relationships among datapoints that go beyond those of simple independent, discrete categories. By using supervised contrastive learning as an additional baseline, we show how training models with such relationships in mind can lead to greater alignment with human behavior. For details on how supervised contrastive learning is implemented in each domain, see Appendix L and Appendix N. Note that we did not include this baseline in the quadrilateral domain since in that case there is already an alternative supervised baseline that has been used in the literature and therefore it is more appropriate (Sablé-Meyer et al., 2021).

In both the Geoclidean (Figure O1) and Natural Image Hierarchy domain

(Figure O2), we see that SupCon can capture pairwise human similarity judgments at or slightly above the level of GenSim. However, GenSim is a clear favorite when predicting human behavior or exhibiting human-like performance on complex tasks such as the Geoclidian task (Figure 8A), where people have to carefully do few-shot classification on image stimuli with distractor stimuli that are generated from highly similar Geoclidian programs (Figure O1C), or the Levels Dataset task (Figure O2A), where people do similarity judgments specifically at different abstraction levels. These results suggest that SupCon has the ability to capture human behavior *prima facie* (e.g. quick task-agnostic similarity judgments) but not enough to capture more complex human behavior during tasks that require reasoning over abstract concepts or hierarchies. Note that, as explained above, SupCon fits within our generative similarity framework using a generative model with discrete and independent categories (or as a Monte-Carlo approximation as noted in the main text). Therefore, this attests to generative similarity’s ability to capture human behavior on a variety of complexities, from *prima facie* similarity judgments to tasks requiring complex reasoning about abstract generative processes, such as the Geoclidian task.

**Figure O1**

Performance in the Geoclidian domain across models with supervised contrastive learning included. **A.** Geoclidian classification accuracy. This is the same as Figure 8C, but with supervised contrastive learning included (black bar). GenSim performs the best at the Geoclidian task. **B.** Similarity scores from all three models vs. human judgments (same as Figure 8B, but with supervised contrastive learning (SupCon) added). Note that for the supervised case the measure of similarity is given by the negative Euclidean distance as this was the measure used in the SupCon triplet-loss specification.

**Figure O2**

Performance in the ImageNet domain across models with supervised contrastive learning included. **A.** (top) ImageNet hierarchy classification performance across different tree levels. This is the same as Figure 9C, but with supervised contrastive learning included (black bars). (bottom left) We aggregated performance across tree levels via the metric $\sum_{i=1}^n \log \frac{a_i}{c_i}$, where a_i is the accuracy at tree level i , c_i is chance accuracy at tree level i , and n is the number of tree levels. (bottom right) Levels dataset choice prediction. This is the same as Figure 9D, but with supervised contrastive learning included (black bars). Overall, we see that GenSim is consistently better than SimCLR and SupCon at capturing human-like capabilities and behavior in reasoning about natural image stimuli at different hierarchical levels. **B.** Similarity scores from all three models vs. human similarity judgments (same as Figure 9, but with supervised contrastive learning added). As with Geoclidean, SupCon similarity is specified as the negative Euclidean distance between embeddings since this was the measure used in the triplet loss training.