Representational Geometry in Neural Networks: From Visual Relational Reasoning to Speech Recognition

A DISSERTATION PRESENTED
BY
JIAQI SHANG
TO
THE DEPARTMENT OF NEUROSCIENCE

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
IN THE SUBJECT OF
NEUROSCIENCE

HARVARD UNIVERSITY
CAMBRIDGE, MASSACHUSETTS

NOVEMBER 2026

©2026 – Jiaqi Shang
all rights reserved.

Thesis advisor: Professor Gabriel Kreiman and Haim Sompolinsky Jiaqi Shang

Representational Geometry in Neural Networks: From Visual Relational Reasoning to Speech Recognition

Abstract

How do neural systems organize their representations so that they can recognize the same category across widely varying instances while preserving distinctions between categories? For abstract relational rules and linguistic units, identity must be recovered across changes in the objects that instantiate a relation or in the acoustic and contextual realization of a linguistic unit. Behavioral or classification accuracy reveals whether a system succeeds, but not which properties of its internal representations make that success possible. This dissertation asks how those properties can be identified and measured.

To answer this question, the dissertation extends a geometric theory developed for perceptual object recognition. In this account, each instance of a category evokes a pattern of activity across a population of neurons or model units, and the set of these responses forms the category's manifold. Separability is measured with a few-shot prototype readout: a few responses from each category are averaged into a prototype, and a new response is assigned to the category of the nearest prototype. The theory predicts this readout from a signal-to-noise ratio (SNR) determined by the separation between manifold centroids and the magnitude, dimensionality, and alignment of within-category variability.

The first study introduces SimplifiedRPM, a controlled relational-reasoning task in which rules combining a relation with an attribute are withheld from training. Across four architectures trained on the same data, the SNR predicts prototype error on these unseen rules. At the same time, its decomposition reveals that the architectures construct rule separability through different layer-wise transformations. Signal and dimensionality trade off through a structured compression that aligns held-out rules with the subspace spanned by the training rules. The geometry is not merely descriptive: directly optimizing the ratio yields lower held-out error than cross-entropy and two alternative geometry-based objectives. The architecture that generalizes best also reproduces the ordering of relational difficulty observed in human participants.

The second study asks whether the same theory describes a large pretrained system processing natural input. Across the processing stages of the speech recognizer Whisper, the SNR again predicts the prototype error of word and phoneme manifolds. Here the decomposition distinguishes mechanisms that accuracy alone would conflate: encoder word separability improves first as within-word variability spreads across more dimensions and later through amplification of the centroid

signal, whereas decoder improvement is dominated by signal amplification. Phoneme manifolds follow a distinct trajectory and peak earlier than word manifolds. Sentence context adds variability that is aligned with word-separating directions at the encoder output but only weakly aligned in the decoder.

Together, the studies show that the geometric account developed for perceptual object recognition extends quantitatively to relational rules, words, and phonemes. Across architectures, processing stages, and category types, the same decomposition predicts the error while revealing the distinct transformations through which each system constructs separability. Because each geometric term can be estimated from neural recordings of responses to many instances of a category, the model findings yield falsifiable predictions for neural populations engaged in relational reasoning and speech perception.

Contents

1 Introduction 1
- 1.1 From sensory input to meaningful structure . . . 1
- 1.2 Population representations of latent structure . . . 2
- 1.3 The gap between information and organization . . . 4
- 1.4 Manifold geometry and few-shot classification . . . 5
- 1.5 Representational geometry in visual reasoning and speech recognition . . . 7
- 1.6 What the dissertation establishes . . . 10
- 1.7 Organization of the dissertation . . . 12

2 Unraveling the Geometry of Visual Relational Reasoning 13
- 2.1 Summary of research in this chapter . . . 14
- 2.2 Background . . . 16
- 2.3 *SimplifiedRPM* enables evaluation of relational generalization . . . 23
- 2.4 SCL aligns with the human relational difficulty structure . . . 30
- 2.5 A geometric framework for relational generalization . . . 34
- 2.6 Layer-wise geometry reveals model-specific generalization strategies . . . 43
- 2.7 Structured compression and the signal–dimensionality trade-off . . . 46
- 2.8 Direct optimization of SNR improves generalization . . . 49
- 2.9 Generality of the geometric account . . . 53
- 2.10 Discussion . . . 57
- 2.11 Methods . . . 63

3 Geometric Signatures of Speech Recognition 69
- 3.1 Summary of research in this chapter . . . 70
- 3.2 Background . . . 72
- 3.3 Extracting word representations from Whisper . . . 79
- 3.4 Word separability across Whisper's depth . . . 83
- 3.5 Geometric mechanisms of word separability . . . 89
- 3.6 Acoustic, phonemic, and semantic differences across depth . . . 92
- 3.7 Phoneme manifolds follow a distinct geometric trajectory . . . 96
- 3.8 Context modulates word geometry in the decoder . . . 102

3.9 Discussion . . . 107
3.10 Methods . . . 116

4 Conclusion 128
4.1 A common geometric account of few-shot separability . . . 128
4.2 Multiple geometric routes to separability . . . 130
4.3 From geometry to behavior and neural populations . . . 132
4.4 Boundaries of the account and the experiments they motivate . . . 134

Appendix A Supporting Material for Chapter 2 139
A.1 The *SimplifiedRPM* dataset . . . 139
A.2 Model architectures and training . . . 148
A.3 Model performance . . . 152
A.4 Evaluation of Qwen-VL on *SimplifiedRPM* . . . 160
A.5 Human experiment . . . 162
A.6 Human–model comparison . . . 165
A.7 Extension towards canonical RPM . . . 170
A.8 Combinatorial rule composition . . . 174
A.9 Rule-dependent geometry in SCL . . . 176
A.10 Structured compression . . . 178
A.11 Training with SNR loss . . . 182

Appendix B Supporting Material for Chapter 3 191
B.1 Stimulus sets . . . 192
B.2 Layer indexing . . . 197
B.3 The prototype signal-to-noise estimator . . . 199
B.4 Supplementary analyses of linguistic features . . . 204
B.5 Supplementary analyses of phoneme manifolds . . . 211
B.6 Supplementary analyses of sentence context . . . 220

References 236

Listing of figures

2.1 The *SimplifiedRPM* dataset and model performance 28
2.2 Example *SimplifiedRPM* rules and rule splits 28
2.3 SCL aligns with the human relational difficulty order 33
2.4 Generalization error is described by relational representation geometry 42
2.5 Layer-wise evolution of generalization geometry across architectures 46
2.6 Structured compression underlies the signal–dimensionality trade-off 49
2.7 Direct optimization of SNR improves relational generalization 53
2.8 The Scattering Compositional Learner architecture 68

3.1 Architecture of the Whisper encoder–decoder model 80
3.2 Linear decodability of word identity across Whisper layers 84
3.3 Few-shot prototype readout of word identity across Whisper layers 85
3.4 Manifold geometry predicts few-shot word error 86
3.5 Few-shot error and signal-to-noise ratio across Whisper layers 88
3.6 Geometric mechanisms of word separability 90
3.7 Acoustic and phonemic differences organize separability at different depths 93
3.8 Sentence context reveals semantic organization in the decoder 95
3.9 Phoneme separability across the Whisper encoder 98
3.10 Geometry of phoneme manifolds across the Whisper encoder 99
3.11 Phoneme-pair separability across the encoder 100
3.12 Effects of preceding context on word separability 104
3.13 Geometric effects of preceding context 106

A.1 Geometry predicts relational generalization under renderer shifts 147
A.2 Relation-wise test performance across models 156
A.3 Layer-wise generalization geometry is robust to random seed 156
A.4 Layer-wise PCA visualization of held-out rule manifolds 159
A.5 Example trials in the human experiment 164
A.6 Model generalization error versus human performance 169
A.7 Zero-shot extension to the canonical 3×3 RPM format 173
A.8 Geometric prediction under combinatorial rule composition 175

A.9 Variance of held-out rules explained by training-rule PCs 181

B.1 Few-shot error against signal-to-noise ratio at every encoder stage 201
B.2 Few-shot error against signal-to-noise ratio at every decoder stage 202
B.3 The bias, bias–variance and noise–noise terms across Whisper layers 203
B.4 Signal term of the matched groups 207
B.5 Input acoustic distance as a target 208
B.6 BERT distance as the semantic measure 209
B.7 Grammatical class 210
B.8 Phoneme-pair separability on the spectrogram, every phoneme labeled 212
B.9 Phoneme-pair separability at encoder layer 9, every phoneme labeled 213
B.10 Phoneme-pair separability at encoder layer 18, every phoneme labeled 214
B.11 Phoneme-pair separability at every encoder layer 215
B.12 Phoneme-pair error against the theory at every encoder layer 216
B.13 Margin of the 39-way phoneme probe across the encoder 217
B.14 The bias–variance and noise–noise terms of the phoneme manifolds 218
B.15 Word organization within phoneme manifolds across the encoder 219
B.16 The three input conditions and the position-matched control 224
B.17 Manifold radius and centroid separation with and without sentence context 225

To S.

Acknowledgments

This dissertation would not have been possible without the guidance and generosity of many people, and I am grateful to each of them.

I am deeply grateful to my advisors, Gabriel Kreiman and Haim Sompolinsky. Gabriel's vision for the field and unwavering support gave me the confidence to pursue new directions, and his generosity as a mentor has been a constant throughout my time in graduate school. Haim's commitment to rigorous and insightful research has shaped the way I think about scientific problems, and I have cherished our many conversations over the years. I could not have asked for a better pair of mentors.

I thank my dissertation advisory committee, Samuel Gershman, Carlos Ponce, and Jan Drugowitsch, whose questions at each meeting sharpened the ideas presented here.

The work in Chapter 3 was carried out with Shailee Jain and Edward F. Chang at the University of California, San Francisco. I have learned much from our discussions, and I am grateful to Shailee for her friendship as well as her collaboration.

I have been fortunate to work alongside wonderful colleagues. In the Kreiman lab, Elisa Pavarino, Mengmi Zhang, Chenguang Li, Dianna Hidalgo, Morgan Talbot, Jie Zheng, Spandan Madan, Leonardo Pollina, Sara Djambasovska, Bastien LeLan, and Yuchen Xiao were generous with their time and their help. In the Sompolinsky group, Qianyi Li, Haozhe Shan, Weishun Zhong, Alexander van Meegen, Lorenzo Tiberi, Kazuki Irie, Xu Pan, Naoki Hiratani, Zechen Zhang, and many others made the lab a place of spirited discussion and good company.

I thank my friends at Harvard, Lily Zhang, Jingxuan Fan, Siyan Zhou, Binxu Wang, Yuchen Xiao, and Albert Chen, for their companionship through these years, and my longtime friends, Shreya Goel, Ruochen Yang, Chuhan Liu, Zibu Liu, Danqi Liao and many others, whose friendship has been a constant across every distance.

The work in this dissertation was supported by the National Institutes of Health grant R01EY026025, the Swartz Foundation, the Gatsby Charitable Foundation, the Office of Naval Research grant No. N0014-23-1-2051, Amazon, and the Kempner Institute for the Study of Natural and Artificial Intelligence at Harvard University.

Finally, I thank my parents and my grandparents for their unconditional love and support.

1

Introduction

1.1 From sensory input to meaningful structure

How do neural systems transform variable sensory inputs into representations that support perception, reasoning, and generalization? The same word, or the same relational rule, reaches the senses as many physically different inputs. Conversely, similar inputs can express different words or relations. A neural system must construct representations that preserve the distinctions relevant for behavior

across changes in their sensory realization.

Speech and visual reasoning pose this problem at different levels. The acoustic form of a phoneme or word varies with speaker, rate, and surrounding sounds, yet listeners recognize the same phonemes and words in new utterances[1,2]. A visual relation, by contrast, is defined over objects rather than contained in any one of them. The same relation can be expressed through different shapes, colors, numbers, or spatial arrangements and recognized in configurations not previously encountered[3]. Speech thus demands generalization to new realizations of known categories; relational reasoning demands transfer of structure across different visual contents.

The common requirement is not invariance to all variation, but sensitivity to which variation matters. Differences among speakers, pronunciations, objects, or configurations may remain represented, provided that they do not obscure the relevant identity or relation. At the same time, differences that change a phoneme, word, or rule must remain distinguishable. The problem for a neural system is to reorganize sensory variation so that task-relevant structure becomes accessible, without collapsing distinct realizations onto a single point.

Behavior reveals the outcome of this reorganization but not how it was achieved. Two systems that recognize the same words or rules equally well may rely on entirely different internal representations. To explain perception and reasoning, one has to examine the transformation itself: how representations change across processing stages, and which of their properties make a category or relation available to subsequent computation. Population activity is the level at which this question can be asked.

1.2 Population representations of latent structure

At each stage of neural processing, an input evokes a pattern of activity across a population of neurons. This population response is the representation available to subsequent stages. Its information

need not reside in neurons selective for individual categories: single neurons often respond to several stimulus and task variables, while distinctions emerge from their joint activity[4]. As signals pass through a neural system, each population transforms the activity produced by the preceding stage.

Sensory hierarchies show this transformation stage by stage. In the primary visual cortex, the population response to an object is dominated by the position, size, and pose of its local features; along the ventral stream, this dependence is progressively reduced[5,6], and in macaque inferotemporal cortex object identity can be read out by a linear classifier across changes in position and scale[7]. In audition, primary auditory cortex represents the spectrotemporal content of sound; in the superior temporal gyrus, population responses are organized by phonetic features, the articulatory and acoustic properties that distinguish speech sounds, in a form that holds across speakers[8,9] and that preserves the order of phonemes within a sequence[2]; and anterior temporal regions respond to word forms[10]. In both systems, sensory processing transforms population activity, stage by stage, into forms that support stable recognition.

Population activity can also encode task variables in forms that generalize across conditions, extending this principle beyond sensory identity to more abstract structure. Prefrontal neurons encode learned categories independently of image similarity[11], and prefrontal and hippocampal populations encode task context and outcome value such that a readout trained under one set of conditions recovers them under another[12].

Together, these findings point to a common computational requirement: categories and relations must remain recoverable from population activity across changes in sensory input and task context. Decoding analyses establish whether this information is accessible, but not how the population makes it accessible. The same readout performance can arise from different organizations of population activity, with different consequences for generalization. The next section therefore turns from whether a variable can be decoded to the representational geometry that supports its decoding.

1.3 The gap between information and organization

Neural representations are often assessed through selectivity and decoding. Selectivity asks whether individual neurons or units respond differently across conditions, whereas decoding asks whether a chosen readout can recover a variable from population activity. These analyses establish that the activity carries information about the variable, but they do not by themselves reveal how that information is organized. In particular, decoding performance depends on the choice of readout, the amount of training data, and the regularization used to fit it. It therefore characterizes the information available to a specified decoder rather than the representational structure that makes decoding possible.

The same decoding performance can arise from different representational organizations. Category means may be widely separated but accompanied by substantial variability along the directions that distinguish them. Alternatively, weaker separation may be offset by smaller or more favorably oriented variability. Accuracy measures the combined consequence of these properties without identifying their individual contributions; near ceiling, it also becomes insensitive to further geometric changes. Explaining how a representation supports classification therefore requires decomposing performance into the properties that produce it.

The geometry of category manifolds provides such a decomposition. One theoretical framework relates linear classification capacity to manifold properties including radius and dimensionality [13]; these quantities have been used to characterize the emergence of object separability across deep-network layers [14]. A complementary framework predicts few-shot prototype-classification error from the separation between category means and the magnitude, dimensionality, and alignment of within-category variability [15]. These approaches make it possible to distinguish representational transformations that produce similar classification performance through different geometric mechanisms.

Most applications of these frameworks have examined object manifolds, whose variability arises from changes such as pose, position, and background. The variables studied in this dissertation pose a broader test. Relations are instantiated across different objects, and words across utterances that differ in speaker, acoustic realization, and context. Their within-category variability may therefore contain both variation that should be discounted and structure that is integral to how the variable is instantiated. This raises two questions. Does the geometric account developed for object manifolds extend to relational and linguistic variables? And can its components reveal how different architectures and processing stages construct representations of those variables?

1.4 Manifold geometry and few-shot classification

At a given processing stage, the response to stimulus i of category a is a vector $\mathbf{x}_i^a \in \mathbb{R}^N$, whose coordinates are the activities of N neurons or units. The set of responses to stimuli belonging to category a defines its manifold [13,15]. Its centroid, $\mathbf{x}_0^a$, is the mean response, and each response can be written as $\mathbf{x}_i^a = \mathbf{x}_0^a + \delta_i^a$, where δ_i^a is its within-category deviation. In the visual task studied in Chapter 2, categories are defined by relational rules: relations such as constancy or progression applied to object attributes such as shape or number. A rule manifold therefore consists of responses to stimuli governed by the same rule. In the speech tasks studied in Chapter 3, a phoneme or word manifold consists of responses to different occurrences of the same linguistic unit.

To relate manifold geometry to readout, pairwise category separability is assessed with a few-shot nearest-prototype classifier. For categories a and b, a small number of reference responses sampled from each are averaged to form a prototype; in the simplest, one-shot case, a single response serves as the prototype. An independent test response is assigned to the nearer prototype in Euclidean distance, and the error is averaged over draws of the reference and test samples. Once the prototypes are specified, their perpendicular bisector defines the decision boundary. This readout differs from a

trained linear probe, whose boundary depends on optimization and regularization, and from manifold capacity, which asks how many manifolds can be linearly separated.

The classifier's error can be predicted from the geometry of the two manifolds[15]. Let R_a^2 denote the mean squared radius of manifold a, D_a its dimensionality, $\mathbf{U}_a$ its principal axes weighted by their normalized radii, and $\Delta\mathbf{x}_0 = (\mathbf{x}_0^a - \mathbf{x}_0^b)/R_a$ the centroid distance in units of R_a. For a test response drawn from category a and classified against category b in the one-shot case, the decision variable is approximately Gaussian. The resulting error is

$$\varepsilon_{a,b} \approx H\big(\mathrm{SNR}_{a,b}\big)\,, \qquad \mathrm{SNR}_{a,b} = \frac{1}{2}\,\frac{\|\Delta\mathbf{x}_0\|^2 + R_b^2/R_a^2 - 1}{\sqrt{D_a^{-1} + \|\Delta\mathbf{x}_0 \cdot \mathbf{U}_b\|^2 + \|\Delta\mathbf{x}_0 \cdot \mathbf{U}_a\|^2}}, \tag{1.1}$$

where $H(x) = \frac{1}{2}\,\mathrm{erfc}(x/\sqrt{2})$ is the Gaussian tail function. Section 2.5 presents this expression in detail for the matching task used in the visual study.

Equation 1.1 separates four contributions. The *signal*, $\|\Delta\mathbf{x}_0\|^2$, is the squared distance between the centroids relative to the radius of the target manifold. The *bias*, $R_b^2/R_a^2 - 1$, is the difference in size between the two manifolds. The *dimensionality* term, D_a^{-1}, is the contribution of within-category variability. The *signal–noise overlap* terms, $\|\Delta\mathbf{x}_0 \cdot \mathbf{U}_a\|^2$ and $\|\Delta\mathbf{x}_0 \cdot \mathbf{U}_b\|^2$, measure how much of the within-category variability lies along the direction that separates the centroids. Error falls with larger signal and dimensionality and rises with overlap. No single term determines it: the benefit of greater centroid separation is reduced when variability aligns with it. When each prototype is the average of m reference stimuli, the bias, dimensionality, and reference-side overlap terms are divided by m, whereas the overlap of the test manifold is unaffected; the m-shot form of the expression is Eq. A.4 in Appendix A.7, and the speech study uses it with $m = 5$.

This decomposition addresses the gap identified in the preceding section. Classification error reports the combined outcome of these four properties; the geometry measures each of them separately and predicts the error from their combination. A change in separability between two pro-

cessing stages can then be attributed to the terms that changed, distinguishing a stage that amplifies the signal from one that expands dimensionality, reduces within-category variability, or rotates that variability away from the discriminating direction. Because the four quantities are defined by the manifolds alone, they can be compared across layers, architectures, and categories of different kinds. One distinction is needed to read them correctly. The dimensionality that enters Eq. 1.1 is that of the variability within a category, not that of the representation as a whole.

The framework rests on three approximations: each manifold is summarized by its centroid and covariance, neglecting higher-order structure; the readout is a Euclidean nearest-prototype rule; and the decision variable is treated as Gaussian. It provides a quantitative and falsifiable relation between the measured geometry of a representation and readout error at any stage where both can be measured. Deep networks offer a tractable setting in which to test this link: the activity of every unit can be measured across successive layers, architectures can be compared under controlled conditions, and representational geometry can be tracked or directly modified.

1.5 Representational geometry in visual reasoning and speech recognition

This section asks whether the geometric account extends to relational rules and linguistic units, and whether its components reveal how representations of them differ across architectures and processing stages. Addressing these questions directly in a nervous system is difficult because it requires measuring population activity across processing stages while controlling stimulus variation and the system's prior experience with the categories. Deep neural networks provide complementary access. Every unit can be observed at every layer, and controlled tasks allow stimuli or categories to be withheld from training. Architectures can also be compared under matched datasets and objectives, allowing their inductive biases to be related to differences in representational geometry. Pretrained models offer less control over training but make it possible to study representations in systems that

perform natural tasks at scale.

Deep networks are relevant to neuroscience without being replicas of the brain. Networks optimized for visual and auditory tasks predict aspects of cortical population responses [16,17], and each quantity in Eq. 1.1 can also be estimated from neural recordings. The networks are used here as tractable systems for developing hypotheses about population computation. Two settings are examined: a controlled visual-reasoning task in which held-out relational rules, alternative architectures, and interventions on training can be compared, and natural speech recognition in a pretrained encoder–decoder hierarchy whose representations can be followed stage by stage.

1.5.1 Abstract relational structure in vision

The first setting is abstract relational structure in vision. Relational reasoning provides a stringent test of generalization because a rule is defined by the structure relating multiple objects rather than by the appearance of any object alone. Two stimuli governed by the same rule need not share any object. In the task studied here, each stimulus is a row of rendered objects, and a rule specifies how one attribute of the objects, such as their number or size, changes along the row; forty rules pair ten relations with four attributes. A rule is presented through a single example row, and the network must identify which of two candidate rows follows the same rule by comparing their representations. This is the one-shot case of the nearest-prototype readout of Section 1.4. Generalization is tested on rules withheld entirely from training. The relation and the attribute of a held-out rule each occur in other training rules, but their combination does not, so no response to a held-out rule has been shaped by training on it. Because the stimuli are synthetic, the varying attributes are specified exactly, and because the architectures are trained from scratch on the same data and objective, differences in geometry and generalization can be attributed to inductive biases.

The first question in this setting is whether the geometric account extends from object categories to relational rules, that is, whether the error on held-out rules follows Eq. 1.1. If it does, the four

terms can be used to ask which of them change as training makes held-out rules separable, and at which layer of each architecture that separability appears. Two further questions are whether training on the signal-to-noise ratio itself reduces held-out error relative to cross-entropy training, and whether the ordering of relational difficulty in the networks matches the ordering of difficulty in human participants on the same stimuli.

1.5.2 Hierarchical linguistic structure in speech

The second setting examines linguistic structure in natural speech. Recognizing a phoneme or word requires recovering its identity across variation in speaker, pronunciation, duration, acoustic context, and surrounding language. The speech signal does not explicitly delimit these units, and phonemes and words pose the problem at different scales: phonemes are brief and locally context-dependent, whereas words extend over sequences of phonemes and depend on their identity, order, and timing. The model studied is Whisper [18], a large pretrained encoder–decoder network for speech transcription. Its encoder transforms the acoustic input through successive stages, while its decoder uses that representation to generate a transcript. A phoneme manifold is formed from occurrences of the same phoneme, and a word manifold from utterances of the same word; word manifolds are followed across the full encoder–decoder hierarchy, and phoneme manifolds across the encoder. Unlike the controlled visual experiment, this setting offers a test of the geometric framework in a system trained at scale to perform a natural task.

The questions in this setting are at which stages of the hierarchy words become separable under the few-shot readout, whether Eq. 1.1 predicts the measured error at each of those stages, and which of the four terms accounts for each change in separability between successive stages. A further question is whether a large acoustic, phonemic, or semantic difference between two words increases their separability, and at which depths each kind of difference does so. Two further questions concern the scale of the linguistic unit and its context: whether phoneme identity is constructed through the

same geometric changes as word identity or through different ones, and how the preceding sentence alters the representation of a word, in the encoder, in the decoder, or in both.

1.6 What the dissertation establishes

This dissertation establishes two general results. First, measured manifold geometry predicts few-shot classification error. In the controlled visual setting, this prediction extends to relation–attribute combinations withheld from training, and directly optimizing the signal-to-noise ratio improves generalization to those combinations; in the speech setting, the same prediction holds across the stages of a pretrained hierarchy. Second, separability is not produced by a monotonic change in any single geometric property. It emerges from joint changes in signal, bias, dimensionality, and signal–noise overlap whose contributions depend on the architecture, processing stage, and variable represented. Similar readout performance can arise through distinct internal transformations that a classification score alone cannot distinguish.

In the visual study, the geometric SNR predicts held-out rule error across four architectures. Directly optimizing this quantity produces lower held-out error than cross-entropy and two alternative geometry-based objectives. Layer-wise decomposition further shows that the architectures reach separability through distinct routes. Across layers, training stages, and learning objectives, signal and dimensionality trade off as held-out rules become increasingly aligned with the subspace spanned by the training rules. The architecture that generalizes best also most closely reproduces the ordering of relational difficulty in human participants.

In the speech encoder, word separability first increases through expansion: within-word variability spreads over more dimensions and turns away from the word-separating directions while signal shows no net change. Later encoder improvement is driven by amplification, and decoder improvement is likewise dominated by amplification. Phoneme manifolds follow a different trajec-

tory and peak before word manifolds. Sentence context reorganizes word geometry differently in the two stacks: at the encoder output it adds large, low-dimensional variability aligned with the word-separating directions, whereas in the decoder it adds high-dimensional variability weakly aligned with those directions.

The dissertation's conceptual contribution is to extend the manifold account of few-shot classification from perceptual object categories to relational rules and linguistic units. Its empirical contributions include a controlled benchmark for measuring generalization to withheld rules across architectures, a comparison of the resulting ordering of difficulty with human participants, and a stage-by-stage account of word geometry across a complete encoder–decoder hierarchy and of phoneme geometry across its encoder for natural speech transcription. Methodologically, the same geometric decomposition predicts classification error in both studies, and in the controlled visual setting the resulting SNR serves as a training objective that improves held-out generalization.

These findings yield two testable predictions for neural populations. If relational difficulty is determined by population geometry as it is in the one architecture that reproduces the human ordering, rule manifolds should show smaller centroid separation for the relations that humans find harder. Recording from prefrontal or parietal populations while participants perform the same task would test this. Second, if the transformations observed in Whisper reflect a general principle of speech processing, recordings across the auditory hierarchy should show early gains in word separability through expansion, with rising dimensionality and falling signal–noise overlap while centroid separation changes little, and later gains for both phonemes and words dominated by rising signal, with phoneme separability peaking before word separability.

These conclusions have a defined scope. The manifold theory retains only centroids and covariance structure, and its predictions concern a fixed few-shot prototype readout. The visual study does not provide a theory of why an architecture arrives at a particular geometry, and the human comparison is limited to the ordering of relational difficulty rather than evidence for shared mech-

anisms. The speech study examines one pretrained model, uses externally supplied phoneme and word boundaries, and analyses read speech rather than spontaneous conversation. No neural activity is measured, so the neural predictions above remain hypotheses. Within these limits, this dissertation quantitatively links category geometry to few-shot classification performance and identifies the distinct geometric transformations by which deep networks make abstract variables separable.

1.7 Organization of the dissertation

The remainder of the dissertation is organized as follows. Chapter 2 presents the visual relational-reasoning study: it introduces *SimplifiedRPM* and the held-out-rule design, compares four architectures and relates their ordering of relational difficulty to that of human participants, develops the few-shot manifold analysis for rule manifolds, follows their geometry across layers and training, and tests the signal-to-noise ratio as a training objective. Chapter 3 presents the speech study: it introduces Whisper, constructs word manifolds at every stage of the encoder–decoder pathway and phoneme manifolds throughout the encoder, tests the geometric prediction, decomposes changes in separability across depth, and examines associations between word geometry and acoustic, phonemic, semantic, and contextual differences. Chapter 4 brings the studies together, outlines the implications for neural population computation, and states limitations and directions for neural and computational tests. Appendices A and B provide supporting analysis details.

2

Unraveling the Geometry of Visual Relational Reasoning

This chapter has been published as: Shang J., Kreiman G., and Sompolinsky H., 2026. Unraveling the geometry of visual relational reasoning. *Scientific Reports*. doi:10.1038/s41598-026-47123-3.

2.1 Summary of research in this chapter

Humans readily generalize abstract relations. A person who understands a relation such as "constant" applies it without effort to a row of objects sharing the same color, the same shape, or the same number, and does so on the first encounter with each new instantiation. Neural networks do not. Models trained on large relational-reasoning benchmarks reach high accuracy on the relations they were trained on and then fail, often dramatically, on relations they have not seen. This gap is the subject of this chapter. Rather than asking which architecture or which training procedure closes it, this chapter asks a representational question: what must be true of a network's internal representations for abstract relational generalization to be possible at all?

The chapter develops a geometric answer. Its first component is a benchmark. Existing relational-reasoning datasets entangle abstract relational structure with multi-row inference, multi-choice elimination, and strategic search, and they typically let the relational rules used at test time overlap with those seen during training, making it impossible to attribute success to genuine relational transfer. The chapter introduces *SimplifiedRPM*, a controlled abstraction of Raven's Progressive Matrices in which each trial requires inferring a single unambiguous relational rule from one sample row and matching it against two choice rows. Forty rules are constructed by applying ten abstract relations, spanning progression, arithmetic, and logical families, to four object attributes. Models are trained on 35 rules and evaluated on 5 strictly held-out rules, so that success requires transfer of abstract relational structure rather than interpolation over familiar rule components. To make model behavior interpretable against human behavior on the same stimuli, behavioral data were also collected from 25 human participants on a subset of the task.

The second component is an evaluation. Four architectures with distinct inductive biases for relational processing—ResNet-50, Wild Relation Network (WReN), Vision Transformer (ViT), and the Scattering Compositional Learner (SCL)—all learn the training rules well below chance error

but differ sharply in their generalization to unseen rules. SCL generalizes best, and its pattern of relational difficulty most closely matches human error rates and response times. This correspondence does not demonstrate that humans and SCL use the same mechanism, particularly given their different learning histories. It does, however, show that the ordering of rule difficulty captured by SCL is also present in human behavior rather than being peculiar to the model.

The third component is a geometric theory of relational generalization. Each relational rule is represented as a *rule manifold*: the set of representations that instantiate the rule while varying in attribute values, object arrangements, and other nuisance factors. This extends a framework developed for few-shot object classification to equivalence classes defined by abstract relational structure. Under a second-order approximation, the discriminability of two rule manifolds is predicted by a signal-to-noise ratio comprising four interpretable quantities: the separation between their centroids (*signal*), their relative sizes (*bias*), the effective dimensionality of within-rule variability, and the *signal–noise overlap* between the centroid difference and the dominant directions of variability. Across all four architectures and 15 held-out rule splits, this ratio predicts measured generalization error.

Applying the theory yields three further findings. First, improvements in rule separability are concentrated at architecture-specific processing stages and arise through different geometric transformations across models. Second, signal and dimensionality trade off across architectures, over training, and between objectives, even though they are geometrically independent. This trade-off reflects a structured compression in which held-out rules become increasingly aligned with the low-dimensional subspace spanned by the training rules. Third, the geometric quantities can be optimized directly. The chapter introduces an SNR loss that maximizes the signal-to-noise ratio between rule manifolds within each mini-batch. It reduces held-out error from 0.153 to 0.096, outperforming cross-entropy and two alternative geometry-based objectives; ablations show that every term is required to impose the full set of geometric constraints.

Together, these results provide a quantitative account of relational generalization in this controlled setting. They identify the geometric transformations underlying performance differences and establish an analytical framework that Chapter 3 applies to a different modality.

2.2 Background

2.2.1 Raven's Progressive Matrices and human relational reasoning

Abstract relational reasoning is the capacity to identify a structural regularity independently of the particular objects or attributes that instantiate it, and to apply that regularity in a new context. This capacity supports generalization across problems that share relational structure but differ in their surface features and is widely regarded as an important component of human intelligence [19,20]. Among the tasks developed to study this capacity, Raven's Progressive Matrices (RPM) [21] has become a widely used paradigm for assessing abstract relational reasoning.

In a typical RPM problem, the participant is presented with a 3×3 matrix of panels in which the bottom-right panel is missing. The participant must infer the relations governing the rows or columns—for example, that shape remains constant or that the number of objects increases—and select the candidate that correctly completes the matrix. Because the relevant information is contained primarily in the relations among panels rather than in domain-specific knowledge, RPM was developed as a relatively culture-fair measure of reasoning ability. The same property makes it well suited to computational study: successful performance requires a model to identify and apply relational structure across changes in the objects and attributes that instantiate it.

Two findings from behavioral studies of RPM are particularly relevant to this chapter. First, performance on the task is strongly associated with measures of general cognitive and problem-solving ability [3]. Second, item difficulty varies systematically across relation types. Progression relations, in which an attribute changes monotonically across a row, are generally easier than arithmetic rela-

tions, which require values from multiple panels to be combined. Logical relations, including AND, OR, and particularly XOR, are typically the most difficult[22,23].

The human–model comparison in this chapter is limited to the relative ordering of difficulty across relation types. Human participants complete RPM without extensive task-specific training, whereas the neural networks are trained on tens of thousands of examples drawn from 35 relational rules. Their absolute accuracies are therefore not directly comparable. Agreement in the ordering of relation difficulty is interpreted as evidence that similar structural properties of the task shape human and model performance, but not as evidence that they acquire or represent those relations through the same mechanisms.

Beyond behavioral evidence, neurophysiological studies have examined how neural populations represent categorical and relational variables abstractly. Population activity in prefrontal and parietal cortex encodes learned categories and task variables across changes in the stimuli that instantiate them[11,24]. In contrast, representations of object identity in inferotemporal cortex exhibit substantial tolerance to changes in appearance[25]. Studies in humans and non-human primates have further identified population representations of abstract, relational, and conceptual structure during tasks that require inference over learned relationships[26,27,28,29]. Together, these findings motivate a representational analysis of relational reasoning. If abstract relations are encoded in patterns of population activity, then the geometry of those patterns provides a natural level at which to investigate the basis of relational generalization.

2.2.2 Computational benchmarks for relational reasoning

Inspired by RPM, the machine-learning community has built large-scale relational-reasoning datasets intended to make the ability measurable and trainable. Two have been especially influential. Procedurally Generated Matrices (PGM)[30] generates RPM-style problems from a formal specification of relations, attributes, and objects, and defines a family of generalization regimes by controlling

which combinations of these appear in training versus test. RAVEN[31] constructs problems with an explicit hierarchical rule grammar and richer figure configurations, and pairs each problem with a structured annotation of its underlying rules.

These datasets made relational reasoning a tractable machine-learning problem, and the models trained on them[30,31,32,33,34,35,36,37,38,39] reach high, sometimes near-ceiling accuracy on the rules they are trained on. But two problems have emerged that limit what such accuracy can be taken to show.

One limitation of these benchmarks is their susceptibility to shortcut solutions. In the original RAVEN dataset, incorrect candidates were generated by perturbing the correct answer, introducing regularities that allowed models to identify the correct candidate from the answer set alone, without considering the problem matrix. High accuracy therefore did not necessarily indicate that a model had learned the intended relational structure. I-RAVEN and stratified variants[32] revised the answer-generation procedure to remove this confound. This case illustrates the importance of verifying that benchmark performance depends on the relational information the task is intended to assess.

The second limitation concerns the overlap between training and test rules. In existing benchmarks, test rules often overlap with training rules. A model trained on *constant shape* and tested on *constant number* must transfer a familiar relation to a new attribute. Suppose both the relation and the attribute have appeared during training, although in different examples; the test requires only their recombination. Performance declines sharply when relational rules are explicitly withheld[40], suggesting that much of the high accuracy reported on standard splits depends on overlap with familiar relational components. Because the degree and form of this overlap are not systematically controlled, these benchmarks do not reveal how much of model performance reflects abstraction of relational structure.

These limitations are not confined to models trained specifically on RPM benchmarks. Large pretrained vision–language models also perform poorly on RAVEN-style problems, despite their

broad training data and capacity for in-context inference[41]. Recent studies have begun to characterize the sources of these failures and to evaluate methods for improving visual relational reasoning[42,43,44,45,46]. Collectively, these results indicate that large-scale pretraining alone does not produce robust relational generalization.

A further limitation is the scarcity of human behavioral data collected on the same stimuli used to evaluate computational models. Although patterns of human difficulty are well established for classical RPM, contemporary benchmarks differ substantially in their visual structure, rule distributions, and generation procedures. The findings from classical RPM therefore cannot be assumed to generalize directly to these datasets. Without matched behavioral measurements, it is not possible to determine whether the difficulty profile of a model resembles that of human participants.

A complementary line of research examines simpler relational tasks, including transitive inference[47,48,49] and same–different judgments[50,51,52]. The restricted structure of these tasks facilitates mechanistic analysis yet leaves open whether the identified mechanisms extend across multiple relation types or to relations instantiated in more complex visual inputs.

SimplifiedRPM, introduced in Section 2.3, combines the relational diversity and image-based inputs of large RPM benchmarks with the experimental control of simpler tasks. Each row instantiates a single unambiguous rule; training and held-out rule sets are strictly disjoint; potential low-level visual shortcuts are systematically evaluated; and human performance is measured on the same trials presented to the models. These properties support a controlled analysis of relational generalization and its underlying representations.

2.2.3 Models with relational inductive biases

The four architectures evaluated in this chapter represent distinct hypotheses about the computations that support relational reasoning. They range from a general-purpose convolutional network with no explicit relational mechanism to architectures that incorporate pairwise comparison, global

self-attention, or shared relational computation across attributes. Comparing them under matched training conditions makes it possible to examine how these inductive biases shape relational generalization and the representations that support it.

ResNet-50[53] provides the general-purpose convolutional baseline. It tests the hypothesis that relational structure can be learned without a specialized relational mechanism, through the same hierarchical feature transformations used to extract visual information.

The Wild Relation Network (WReN)[30] instead imposes an explicitly pairwise structure. A shared convolutional module encodes each panel, and the same relation function processes every pair of panel representations before the resulting outputs are aggregated. WReN therefore tests the hypothesis that relational reasoning is facilitated by building pairwise comparison directly into the architecture.

The Vision Transformer (ViT)[54] provides a less structured mechanism for integrating information across the input. Visual information is represented as tokens and processed by successive self-attention blocks, allowing each token to interact with every other token. Unlike WReN, ViT does not specify which comparisons should be computed; the interactions relevant to a relation must be learned from the training data.

The Scattering Compositional Learner (SCL)[33] introduces a different inductive bias: relational computations are shared across attributes. SCL decomposes each panel representation into attribute-specific components and applies the same relational module to each component. By separating relational computation from attribute identity, the architecture is designed to support generalization to relation–attribute combinations that were not observed during training.

These four architectures do not exhaust the computational approaches to relational reasoning. Other models incorporate object-centric decomposition[55,39], emergent symbolic or indirection mechanisms[56,57], neuro-symbolic components[35], dynamic or memory-based computation[36,58,59,60], or additional computation at inference time[61,62,63,64,65]. A parallel effort relates such

mechanisms to neural data[66,67]. The present analysis focuses on the distributed vector representations produced during a single forward pass, for which centroids, covariances, and other geometric quantities are well defined. Explicit symbolic and probabilistic accounts that perform inference over structured rule representations[68,69] are not examined here; relating the geometry of learned distributed representations to such inference mechanisms remains an important direction for future work.

2.2.4 Representation geometry and the manifold view of generalization

The manifold view of neural representations begins from the observation that a category is represented not by a single point but by a set of activity patterns. Different images of the same object evoke different responses as pose, lighting, position, and other stimulus properties vary; together, these responses form a *perceptual manifold* in representation space. Processing across a sensory hierarchy can transform these manifolds so that variation within an object does not obscure differences between objects, thereby making them more easily separable by downstream readouts. The theoretical framework applied in this chapter, and subsequently in Chapter 3, provides a quantitative account of this process.

The theory of linear classification of manifolds[13,14] quantifies representational separability through *manifold capacity*: the number of category manifolds that a linear classifier can separate. This capacity depends on two geometric properties of within-category variability: effective radius, which measures its magnitude, and effective dimension, which measures the number of directions over which it is distributed. In this setting, smaller and lower-dimensional manifolds support greater classification capacity. Both quantities can be estimated from population responses, with effective dimensionality quantified from the covariance spectrum using the participation ratio[70].

The framework adapted in this chapter extends the manifold approach from classification capacity to few-shot generalization[15]. Each category is represented by a prototype estimated from a small

number of examples, and new instances are classified according to their distance from competing prototypes. Under a second-order approximation, the error between two manifolds is predicted by a signal-to-noise ratio (SNR) with four geometric components: the separation between manifold centroids (*signal*), a *bias* arising from differences in manifold size, the effective *dimensionality* of within-manifold variability, and the *signal–noise overlap* between the centroid-separation direction and the principal directions of variability. Dimensionality has a different effect here than in the classification-capacity setting. When the total magnitude of within-manifold variability is held fixed, distributing that variability across more independent directions reduces its projection onto any particular decision direction and improves prototype generalization. This quantity describes variability within a category, not the dimensionality of the representation as a whole. Together, the four geometric terms determine the SNR, with higher values predicting lower generalization error.

This chapter extends the framework by redefining the category represented by each manifold. A *rule manifold* is the set of representations of all rows that instantiate the same relational rule across variations in attribute values, object configurations, and other rule-preserving factors. Within-manifold variability therefore captures the different visual realizations of a common relational structure rather than different appearances of a common object. The central question is whether the same geometric quantities that predict generalization across object categories also predict generalization across abstract relational rules.

2.2.5 Open questions and organization of this chapter

Prior work leaves unresolved how networks represent relations in a form that generalizes to rules withheld from training. Existing benchmarks often permit train–test overlap, model comparisons rely primarily on aggregate accuracy, and matched human data are scarce. Geometric theories developed for object categories have not been tested on abstract relations. This chapter asks which representational properties predict relational generalization, how different architectures construct

them, whether they account for human patterns of difficulty, and whether they can be optimized directly.

The chapter proceeds as follows. Section 2.3 introduces the *SimplifiedRPM* benchmark and reports how the four architectures perform on held-out rules. Section 2.4 reports the human behavioral experiment and compares relational difficulty orderings between humans and models. Section 2.5 develops the geometric framework for rule manifolds and tests whether it predicts generalization error. Section 2.6 traces the emergence of relational geometry across network depth. Section 2.7 characterizes the signal–dimensionality trade-off and the structured compression underlying it. Section 2.8 turns the theory into a training objective. Section 2.9 reports three stress tests that relax the benchmark's controls — renderer shifts, the canonical eight-choice RPM format, and combinatorially composed rules — and identifies what distinguishes the geometry of the best-generalizing architecture. Section 2.10 discusses implications and limitations, and Section 2.11 gives methodological detail.

2.3 *SimplifiedRPM* enables evaluation of relational generalization

2.3.1 Task design

To assess abstract relational generalization beyond training experience, this chapter introduces *SimplifiedRPM*, a principled abstraction of Raven's Progressive Matrices[21]. The classic RPM task probes abstract relational reasoning but entangles multiple sources of complexity, including multi-row inference, elimination among multiple answer choices, and strategic search over candidate rules. *SimplifiedRPM* preserves the core requirement of inferring an abstract relational rule from visual examples while removing these strategy-level confounds. Each trial requires inferring a rule from a single *sample row* of three panels and identifying the matching rule from two *choice rows* (Fig. 2.1a), thereby isolating relational matching under controlled conditions.

The dataset comprises rows governed by 40 distinct relational rules. Each rule is defined by applying an abstract relation to a specific object attribute, yielding a single, unambiguous relational structure per row. Ten abstract relations span three relation families: progression (constant, progression ± 1, progression ± 2), arithmetic (addition, subtraction), and logical (AND, OR, XOR). Each relation specifies how attribute index values assigned to objects evolve across the three panels of a row; formal symbolic definitions and illustrative examples for all ten are given in Table 2.1, and rendered examples in Fig. 2.2. This set of relations and attributes was chosen to be consistent with those used in human RPM studies while spanning the progression, arithmetic, and logical families whose difficulty ordering is documented in the behavioral literature (Section 2.2.1). For each rule, many visually distinct rows are generated by varying task-irrelevant attributes.

Each relation is applied to one of four object attributes: shape, size, shade, or number/position (Fig. 2.2b). Number and position are deliberately grouped as a single attribute to avoid rule ambiguity: a constant-position rule inherently fixes the number of objects, thereby enforcing a constant-number relation as well and violating the requirement that each row instantiate exactly one relational rule. To maintain consistency with prior benchmarks, object definitions and attribute specifications follow those of the PGM dataset [30]. Shape is selected from seven geometric forms: circle, triangle, square, pentagon, hexagon, octagon, and star. Size is defined by the height of an object's bounding box, discretized into ten evenly spaced levels from 20 to 40 pixels. Shade is grayscale intensity, discretized into ten levels from white ($[230, 230, 230]$) to black ($[0, 0, 0]$) in RGB space. The number attribute specifies how many objects appear in a panel (from 1 to 9), and position specifies their spatial arrangement. Panels are rendered at 160×160 pixels with objects placed at nine predefined locations on a 3×3 grid, so that each dataset instance has dimensions $3 \times 160 \times 160$ with the three panels presented horizontally.

Rows are generated by a structured sampling procedure that enforces the one-rule-per-row constraint. For a given rule, attribute values are first sampled so that the target relation is satisfied across

the three panels; values for all remaining, task-irrelevant attributes are then sampled independently at random. To prevent unintended relational regularities, a row is rejected and resampled if any non-target attribute inadvertently satisfies a different relational rule. This procedure guarantees a single unambiguous relation per row while producing substantial visual variability across instances of the same rule. For each of the 40 rules, 10,000 training rows, 1,000 validation rows, and 1,000 test rows were generated, giving 480,000 rows in total.

Because a benchmark of this kind is only as good as its resistance to shortcut solutions, an audit was conducted to determine whether low-level image statistics could support task performance in place of relational structure. For each panel, we measured foreground occupancy, edge density, mean luminance, and luminance contrast. Rule identity had statistically significant but very small effects on all of these, explaining only 0.6–3% of the variance, and multinomial classifiers trained on the features alone performed near chance (classification error 0.88–0.94 against a chance level of 0.975). A feature-only matching baseline, which selects the choice row whose normalized feature vector is closest to the sample row, achieved a generalization error of 0.44 ± 0.08 — worse than the weakest image-based model evaluated here. Full details of the audit are given in Appendix A.1.

2.3.2 Held-out rule protocol and model evaluation

Relational generalization is assessed using a strict held-out rule split in which models are trained on 35 of the 40 relational rules and evaluated on the remaining 5 unseen rules. The held-out set is constructed such that each of the five abstract relations appears exactly once. For example, a model trained on "constant number" and "OR shape" does not see "constant shape" during training and is evaluated on it only as a held-out rule. This design enforces true relational generalization by requiring transfer of abstract relational structure rather than interpolation over familiar rule components. In contrast, existing RPM benchmarks[31,30] often permit overlap between training and test rules, obscuring the source of generalization.

Four representative neural architectures are evaluated with distinct inductive biases for relational processing: ResNet-50[53], a feedforward convolutional network; WReN[30], which explicitly aggregates pairwise relations; Vision Transformer (ViT)[54], which models global interactions via self-attention; and the Scattering Compositional Learner (SCL)[33], which employs a shared relational module to promote attribute-invariant reasoning. Section 2.2.3. Each model maps a *SimplifiedRPM* row to a high-dimensional *relational representation* of its underlying rule and is trained end-to-end with a linear classification head using cross-entropy loss.

At evaluation time, relational generalization is assessed by nearest-neighbor matching between the representations of the sample and choice rows (Fig. 2.1b). A prediction is considered correct if the representation of the sample row is closer, in Euclidean distance, to that of the rule-consistent choice than to the alternative. Generalization error is defined as the fraction of incorrect matches on held-out rules in this two-choice *SimplifiedRPM* task. The critical property of this evaluation is that it never trains a readout on the held-out rules: performance depends entirely on how the representation space is organized, which is what makes the geometric analysis in Section 2.5 applicable.

2.3.3 Models learn the training rules but differ sharply in generalization

Across 15 random training and held-out rule splits, all models successfully learned the training rules, achieving test errors well below chance level (Fig. 2.1c). Evaluation on held-out rules revealed large and consistent differences in generalization performance. Although all models outperformed the chance level of 0.5, SCL achieved the lowest generalization error (0.153 ± 0.019, 95% CI), followed by ViT, with WReN and ResNet performing substantially worse (Fig. 2.1d). This ordering is consistent with prior results on the PGM benchmark's held-out rule split[40]. Full statistical details are reported in Appendix A.3.

The gap between the two panels of this comparison is the phenomenon the rest of the chapter explains. Test error on trained rules is far below chance for every architecture, and carries essentially

no information about which model will transfer; generalization error on held-out rules spans a wide range and orders the models unambiguously. Whatever distinguishes SCL from ResNet is therefore not the capacity to fit relational rules, which all four possess, but something about how the fitted representations are organized.

Under the same protocol, Qwen3-VL-8B-Instruct achieved a generalization error of 0.446 ± 0.034 (95% CI; see Appendix A.4), close to the chance level of 0.5. Task-specific fine-tuning of large multimodal models is beyond the scope of this study and remains for future work.

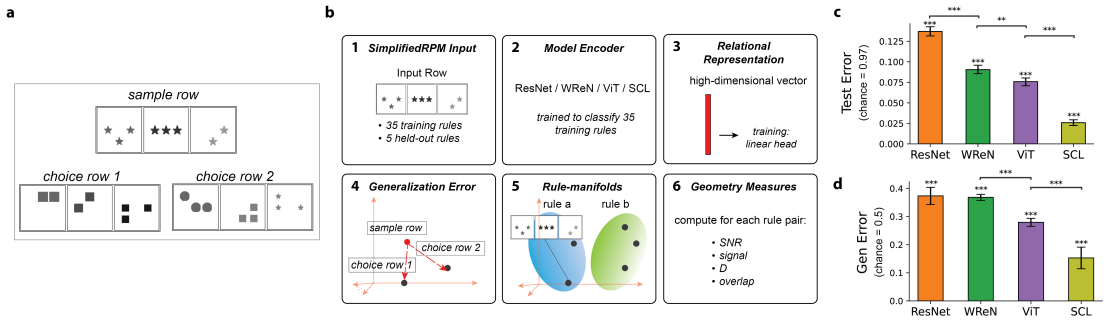


Figure 2.1: The *SimplifiedRPM* dataset enables controlled evaluation of relational generalization in neural networks. (a) Example *SimplifiedRPM* trial. A relational rule is inferred from one sample row and matched to the correct choice. Here, the rule is constant shape: choice 1 is correct, while choice 2 follows constant number. (b) Algorithm flow. Input rows (1) are encoded by model encoders (2) into high-dimensional relational representations (3), trained end-to-end with a linear classifier. Generalization is evaluated by comparing representational distances between sample and choice rows (4). Rule manifolds are representations of the same rule (5), enabling geometric analyses (6). (c) Test error and (d) generalization error on *SimplifiedRPM* for four models, averaged across 15 random splits of training and held-out rules (mean $\pm$ 95% CI). Each split evaluates 20 held-out rule pairs with 500 trials per pair (total: $15 \times 20 \times 500$). All models perform significantly above chance on the test set (one-sample t-test). Asterisks denote significant pairwise differences between models based on Wilcoxon signed-rank tests with Benjamini–Hochberg correction, with FDR-adjusted q-values: ** ($q < 0.01$), *** ($q < 0.001$).

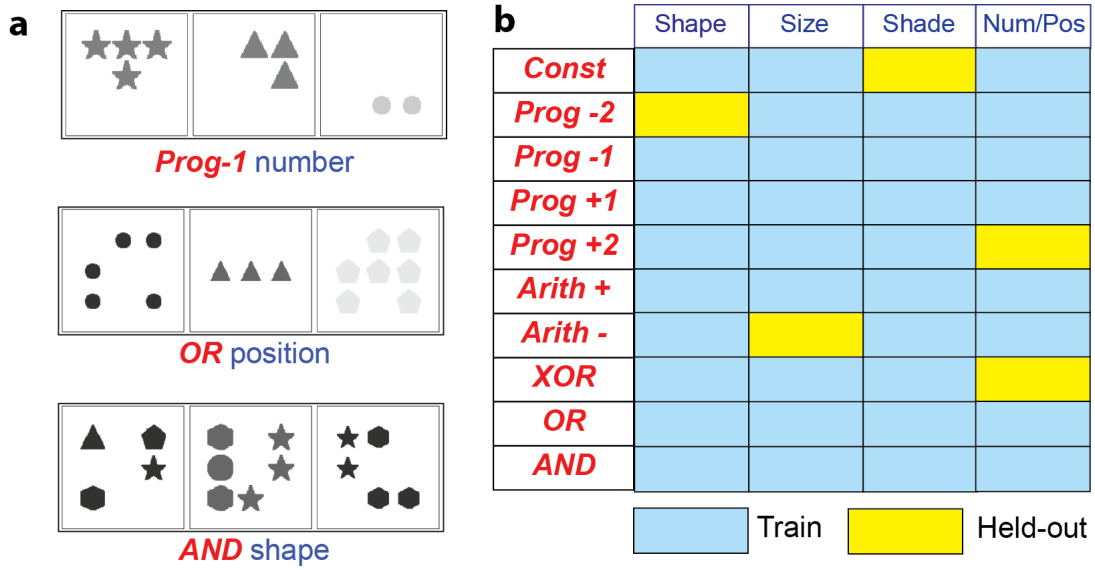


Figure 2.2: Example rules and rule splits in *SimplifiedRPM*. (a) Rendered examples of rows instantiating different abstract relations. (b) The four object attributes to which relations are applied — shape, size, shade, and number/position — and an example split of the 40 resulting rules into 35 training rules and 5 held-out rules, constructed so that each abstract relation appears at most once in the held-out set.

Table 2.1: Relations in the *SimplifiedRPM* dataset. The dataset includes ten abstract relations organized into progression, arithmetic, and logical types. Each rule defines how attribute index values assigned to objects change across three panels. Each example consists of three lists, each representing a panel and specifying the assigned attribute values. For instance, $\{[1], [1], [1]\}$ represents a constant relation, with objects maintaining the attribute value one across all panels. Objects can also have multiple attribute values, as seen in the *AND* relation example $\{[1], [0, 1], [1]\}$, where the second panel contains objects with attributes 0 and 1.

Relation Type	Relation	Description	Example
Progression	Const	Attribute values remain constant across panels.	{[1], [1], [1]}
	Prog -2	Attribute values decrease by 2 across panels.	{[4], [2], [0]}
	Prog -1	Attribute values decrease by 1 across panels.	{[2], [1], [0]}
	Prog +1	Attribute values increase by 1 across panels.	{[0], [1], [2]}
	Prog +2	Attribute values increase by 2 across panels.	{[0], [2], [4]}
Arithmetic	+	Third-panel values equal the sum of the first two panels.	{[0], [1], [1]}
	-	Third-panel values equal the first panel minus the second.	{[1], [1], [0]}
Logical	AND	Third panel contains values present in both preceding panels.	{[0], [0, 1], [0]}
	OR	Third panel contains values present in either preceding panel.	{[0], [0, 1], [0, 1]}
	XOR	Third panel contains values present in only one preceding panel.	{[0], [0, 1], [1]}

2.4 SCL aligns with the human relational difficulty structure

2.4.1 Relation-wise difficulty in models

Performance is first examined separately for progression, arithmetic, and logical relations. This relation-wise analysis reveals distinct difficulty profiles across models (Fig. 2.3a–d): WReN and ViT exhibit their highest generalization errors on arithmetic relations, whereas SCL performs best on progression relations. In ResNet, WReN and ViT test errors increase systematically from progression to arithmetic to logical relations, whereas SCL departs from this ordering on the test set, with its lowest test error on logical relations (Appendix A.3); the generalization differences are therefore not attributable to rule-learning failures. Control analyses further indicate that these relation-wise differences are not explained by rule-dependent variability in attribute sampling or low-level image statistics; if anything, the harder rules are associated with *lower* perceptual variability, the opposite of what a perceptual-load account would predict (Appendix A.3.6).

2.4.2 Human behavioural experiment

Human performance on Raven's Progressive Matrices follows a robust hierarchy of relational difficulty, with progression rules typically among the easiest and logical rules among the most difficult[3,22,23]. By design, RPM testing evaluates human participants without prior task-specific training, making direct comparisons with trained models asymmetric in learning history[71,72]. Accordingly, rather than interpreting human–model similarity as evidence of shared learning mechanisms, the analysis focuses on whether models reproduce the relative ordering of relational difficulty observed in human behavior.

To test this alignment under controlled conditions, we conducted a human behavioral experiment in the *SimplifiedRPM* task with 25 participants and 13 abstract rule pairs. Human partici-

pants performed well above chance, with a mean task error of 0.270 ± 0.038 (mean $\pm$ s.e.m.), and their performance varied systematically across rule pairs, revealing a consistent ordering of relational difficulty. SCL was evaluated on the same *SimplifiedRPM* trials used in the human experiment and showed a comparable mean task error of 0.268 ± 0.043 (mean $\pm$ s.e.m.).

The behavioral data themselves are informative before any model comparison. Table 2.2 reports human error and response time for each of the 13 rule pairs. Difficulty spans nearly the full available range, from 0.11 for discriminating "progression −1 number" from "AND shape" to 0.51 — indistinguishable from chance — for "XOR position" against "arithmetic − number." The ordering is orderly rather than idiosyncratic: pairs in which the correct rule is a progression relation over number are reliably easy, and pairs in which both members are logical relations over the same attribute are reliably hard. Response times run from roughly 6 to 21 seconds and track the error ordering closely, so the two behavioral measures are not independent, but neither is redundant: the hardest pairs by response time are not in every case the hardest by accuracy.

Across rule pairs, SCL's generalization error tracked human error rates (Fig. 2.3e). This alignment extended to response times: rule pairs associated with longer human response times were also those for which SCL exhibited higher generalization error (Fig. 2.3f). Together, these results indicate that SCL reproduces the human relational difficulty order across both accuracy and response time.

That the correspondence holds for response time as well as accuracy is worth noting, since response time was not used in constructing the task, the model, or the geometric analysis, and provides an independent behavioral signature of relational difficulty. It is also the more surprising of the two correspondences. A feedforward network has no analog of deliberation time; its generalization error on a rule pair is a static property of how two manifolds sit in representation space. That this static quantity predicts how long humans take to respond suggests that what makes a rule pair slow for a human and what makes it error-prone for a model may be the same underlying property

of the relational structure — how distinguishable the two relations are, once represented — rather than two separate facts about two separate systems.

Three caveats bound the claim. First, with thirteen rule pairs the error correlation is moderate and its bootstrap interval wide ($\rho = 0.62$, 95% CI $[0.03,\ 0.89]$), whereas the response-time correlation is strong ($\rho = 0.91$, CI $[0.70,\ 0.98]$); the ordering claim rests on both measures together and on the robustness analyses of Appendix A.6. Second, as set out in Section 2.2.1, the comparison is restricted to relative ordering; the near-equality of the mean error rates (0.270 for humans, 0.268 for SCL) is a coincidence of the particular rule pairs selected and carries no weight. Third, the human experiment deliberately excluded rules based on subtle shade differences to minimize purely perceptual difficulty, which means the 13 pairs tested are not a random sample of the 40-rule space. Appendix A.6 provides controls for residual perceptual difficulty, response-time normalization, and an analysis of the robustness of the difficulty ordering to rule-pair resampling; the correspondence survives all three.

Table 2.2: Human performance on the 13 *SimplifiedRPM* rule pairs used in the behavioral experiment. For each pair, the correct rule is the one governing the sample row and the incorrect rule governs the distractor choice row. Values are mean ± s.e.m. across participants.

#	Correct Rule	Incorrect Rule	Human Error	Response Time (sec)
1	Prog-1 Number	AND Shape	0.11 ± 0.04	10.32 ± 1.33
2	Prog+1 Number	Const Number	0.14 ± 0.05	7.23 ± 1.13
3	OR Shape	Arith+ Number	0.15 ± 0.03	13.52 ± 1.08
4	Prog+2 Number	Const Number	0.18 ± 0.06	10.08 ± 1.45
5	Arith- Number	OR Shape	0.18 ± 0.06	13.19 ± 1.46
6	XOR Shape	AND Position	0.20 ± 0.04	18.57 ± 1.57
7	Prog-2 Number	XOR Position	0.22 ± 0.06	8.87 ± 1.23
8	Arith+ Number	OR Position	0.22 ± 0.06	12.13 ± 2.01
9	Const Number	Prog+2 Number	0.23 ± 0.04	6.08 ± 0.58
10	AND Position	Prog-2 Number	0.43 ± 0.07	14.95 ± 1.05
11	AND Shape	XOR Shape	0.47 ± 0.06	17.80 ± 1.07
12	OR Position	XOR Position	0.48 ± 0.07	20.62 ± 1.18
13	XOR Position	Arith- Number	0.51 ± 0.07	14.71 ± 1.02

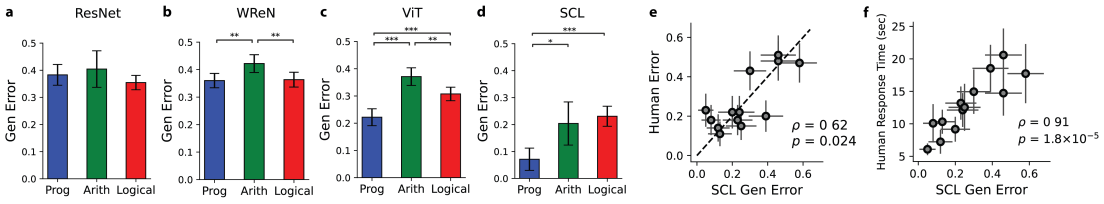


Figure 2.3: SCL aligns with the human relational difficulty order. (a–d) Generalization error for ResNet, WReN, ViT, and SCL evaluated separately on progression (blue), arithmetic (green), and logical (red) relations in the *SimplifiedRPM* task. Bars show mean ± 95% CI, computed over all held-out rule pairs whose target rule belongs to the given abstract relation, aggregated across 15 training/held-out rule splits (500 trials per rule pair). Pairwise differences are assessed using two-sided Mann–Whitney U tests with Benjamini–Hochberg false discovery rate correction ($^{*}\, q < 0.05$, $^{**}\, q < 0.01$, $^{***}\, q < 0.001$). (e–f) Across 13 rule pairs, SCL generalization error correlates with both human error rates (Spearman $\rho = 0.62, p = 0.024$; 95% bootstrap CI $[0.03,\ 0.89]$) and response times (Spearman $\rho = 0.91$, $p = 1.8 \times 10^{-5}$; 95% bootstrap CI $[0.70,\ 0.98]$). Bootstrap confidence intervals estimated from 10,000 resamples of rule pairs. Each point corresponds to one rule pair, based on its 10 trials, each answered by about 10 participants. Error bars indicate 95% confidence intervals.

2.5 A geometric framework for relational generalization

The results so far establish that architectures differ substantially in relational generalization while being indistinguishable in their ability to fit the training rules. To explain that difference, the analysis turns to the structure of the learned representations themselves. This section develops the geometric framework in full — rule manifolds, their second-order description, and the signal-to-noise ratio that governs discriminability — and then tests whether it predicts measured generalization error.

2.5.1 Rule manifolds

Consider a neural network that maps each input row instantiating a relational rule to a representation vector $\mathbf{x} \in \mathbb{R}^N$ at a fixed layer. For a given relational rule a, define the *rule manifold* $\mathcal{M}_a$ as the set of representations corresponding to all rows that instantiate rule a across variations in object attributes, spatial configurations, and other nuisance factors:

$$\mathcal{M}_a = \{\mathbf{x}_{a,i}\}_{i=1}^{P_a}, \tag{2.1}$$

where P_a denotes the number of sampled row instances of rule a used to characterize the manifold.

This definition is the point at which the framework departs from its origins. In the setting of object classification from which the machinery is drawn, a manifold collects the representations of a single object under nuisance variation in appearance, and the semantic content of the category is fixed while its surface form varies [15,13]. A rule manifold is instead an equivalence class over abstract relational structure: its members share no object, no attribute value, and often no visual feature, and are grouped solely by the relation holding among their panels (Fig. 2.4a). The within-manifold variability is not incidental appearance noise. Still, the full space of ways a relation can be realized, and

invariance to it, is exactly what relational abstraction consists of. Whether the geometric description that works for object manifolds also works for equivalence classes of this kind is an empirical question, addressed in Section 2.5.5.

The *centroid* of rule manifold a is

$$\mathbf{x}_0^a = \frac{1}{P_a} \sum_{i=1}^{P_a} \mathbf{x}_{a,i}, \tag{2.2}$$

which captures the mean representation associated with the relational structure of rule a. The covariance matrix characterizes variability within the rule manifold

$$\Sigma_a = \frac{1}{P_a} \sum_{i=1}^{P_a} (\mathbf{x}_{a,i} - \mathbf{x}_0^a)(\mathbf{x}_{a,i} - \mathbf{x}_0^a)^\top. \tag{2.3}$$

Let $\{\lambda_{a,k}, \mathbf{u}_{a,k}\}_{k=1}^N$ denote the eigenvalues and corresponding orthonormal eigenvectors of Σ_a. Following the manifold classification framework[15], the $\mathbf{u}_{a,k}$ define the principal axes of the rule manifold, and $\sqrt{\lambda_{a,k}}$ is its radius along axis $\mathbf{u}_{a,k}$.

The squared radius of the manifold, the mean squared distance of its members from the centroid, summarizes its overall scale,

$$R_a^2 = \operatorname{tr} \Sigma_a = \sum_{k=1}^{N} \lambda_{a,k}, \tag{2.4}$$

and its *dimensionality* is quantified by the participation ratio

$$D_a = \frac{\left(\sum_{k=1}^N \lambda_{a,k}\right)^2}{\sum_{k=1}^N \lambda_{a,k}^2}, \tag{2.5}$$

which measures the number of independent directions of variability the manifold effectively occupies. The participation ratio equals N when variance is spread uniformly over all N directions and approaches 1 when a single direction dominates, so it captures how concentrated the within-

rule variability is rather than merely how large. Throughout the analysis, centroid separations and projection magnitudes are normalized by the manifold radius to ensure scale invariance and comparability across rules and models. Table 2.3 summarizes all geometric quantities and notation used in this chapter.

2.5.2 The signal-to-noise ratio governing relational generalization

Generalization performance is evaluated for *ordered pairs* of rules (a, b), reflecting the asymmetry inherent in discriminating a target rule a from a competing rule b, as in the theory of linear classification of manifolds [15]. Define the normalized centroid separation

$$\Delta \mathbf{x}_0 = \frac{\mathbf{x}_0^a - \mathbf{x}_0^b}{R_a}. \tag{2.6}$$

The signal-to-noise ratio governing discriminability of rule a against rule b is then

$$\mathrm{SNR}_{a,b} = \frac{1}{2} \frac{\|\Delta \mathbf{x}_0\|^2 + R_b^2/R_a^2 - 1}{\sqrt{D_a^{-1} + \|\Delta \mathbf{x}_0 \cdot \mathbf{U}_b\|^2 + \|\Delta \mathbf{x}_0 \cdot \mathbf{U}_a\|^2}}, \tag{2.7}$$

where $\mathbf{U}_a$ and $\mathbf{U}_b$ denote matrices whose columns are the principal axes scaled by the corresponding radii in units of R_a, $\mathbf{u}_{a,k}\sqrt{\lambda_{a,k}}/R_a$ and $\mathbf{u}_{b,k}\sqrt{\lambda_{b,k}}/R_a$ respectively, so that the projections $\Delta \mathbf{x}_0 \cdot \mathbf{U}_a$ and $\Delta \mathbf{x}_0 \cdot \mathbf{U}_b$ are dimensionless. Equation 2.7 shows the dominant terms of the ratio. The values reported in this chapter were computed from the full expression, which carries two further second-order variance terms in the denominator, a bias–variance term and a noise–noise overlap term; both are derived in the supplementary material of Sorscher et al. [15] and are written out for the m-shot readout in Appendix B.3, whose dominant terms at $m = 1$ give Eq. 2.7.

Assuming rule manifolds are high-dimensional, the signed decision margin separating the two rule manifolds is approximately Gaussian, with mean and standard deviation that define a stan-

dardized margin given by $\mathrm{SNR}_{a,b}$. Consequently, the generalization error for a binary decision is determined by the probability that this margin is negative:

$$\varepsilon_{a,b} = H(\mathrm{SNR}_{a,b}), \tag{2.8}$$

where $H(x) = \frac{1}{2}\,\mathrm{erfc}(x/\sqrt{2})$ denotes the Gaussian tail function. This approximation is accurate in the regime where the SNR is $\mathcal{O}(1)$ and manifold dimensionality is large, as validated numerically for both neural data and deep network representations; a full derivation, including the precise conditions under which the Gaussian approximation holds and corrections outside this regime, is given by Sorscher and colleagues [15].

2.5.3 From the matching rule to a decision variable

The nearest-neighbor matching procedure used here can be expressed as a binary decision variable to which the same signal-to-noise analysis applies. In each trial, a sample row is compared with two choice rows, and the choice whose representation lies closer to the sample is selected.

Consider a trial in which the sample row $\mathbf{x}_s$ instantiates rule a, the correct choice row $\mathbf{x}_a$ instantiates the same rule, and the incorrect choice row $\mathbf{x}_b$ instantiates rule b. The model responds correctly when $\|\mathbf{x}_s - \mathbf{x}_a\| < \|\mathbf{x}_s - \mathbf{x}_b\|$, so the signed decision variable is

$$\Delta = \|\mathbf{x}_s - \mathbf{x}_b\|^2 - \|\mathbf{x}_s - \mathbf{x}_a\|^2, \tag{2.9}$$

with a correct response whenever $\Delta > 0$. Expanding the squares, the terms in $\|\mathbf{x}_s\|^2$ cancel and

$$\Delta = 2\,\mathbf{x}_s \cdot \mathbf{w} - \theta, \qquad \mathbf{w} = \mathbf{x}_a - \mathbf{x}_b, \qquad \theta = \|\mathbf{x}_a\|^2 - \|\mathbf{x}_b\|^2. \tag{2.10}$$

Thus, nearest-neighbor matching is equivalent to a linear one-shot prototype classifier: one example

from each rule determines $\mathbf{w}$ and θ, and the resulting classifier is applied to the sample row. Performance therefore depends on how reliably classifiers constructed from single examples separate the rule manifolds, rather than on their separability under an optimally trained readout [15].

2.5.4 Interpretation of the four geometric components

Equation 2.7 decomposes relational discriminability into four interpretable quantities, and much of the analysis in the remainder of this chapter consists of asking which of them a given architecture, layer, or training objective is exploiting.

The *signal* term $\|\Delta\mathbf{x}_0\|^2$ reflects the separation between the centroids of the two rule manifolds, measured in units of the target manifold's radius. Larger centroid separation corresponds to higher $\mathrm{SNR}_{a,b}$ and therefore lower predicted generalization error. This is the most intuitive of the four terms and the one that most naturally corresponds to the informal notion of a representation "distinguishing" two rules.

The *bias* term $\left(R_b^2/R_a^2 - 1\right)$ captures differences in overall manifold scale. When the target rule manifold a is larger than the competing manifold b, this term becomes negative, reducing $\mathrm{SNR}_{a,b}$ and predicting poorer generalization for rule a. The asymmetry of this term is why the framework is defined over ordered rather than unordered rule pairs: a large, diffuse target manifold is easily confused with a compact competitor even when the reverse discrimination is easy.

The *dimensionality* term D_a^{-1} quantifies the effective number of independent directions of variability occupied by the rule-a manifold. Higher dimensionality improves noise averaging along the decision axis and is therefore favored for minimizing generalization error.

Finally, the *signal–noise overlap* terms $\|\Delta\mathbf{x}_0 \cdot \mathbf{U}_a\|^2$ and $\|\Delta\mathbf{x}_0 \cdot \mathbf{U}_b\|^2$ measure the alignment between the centroid separation direction and the dominant axes of variability within each manifold. Greater alignment places more within-rule variability directly along the discriminating direction, reducing $\mathrm{SNR}_{a,b}$, whereas smaller overlap leads to more robust generalization. Two rule manifolds

can therefore be well separated in the sense of having distant centroids and still be poorly discriminable, if the direction that separates them is also the direction along which each rule varies most.

Taken together: rule manifolds are well separated with large SNR when the signal is large, the target manifold is no larger than its competitor, the dimensionality is high, and the signal–noise overlap is minimal.

2.5.5 Geometry predicts generalization error

To test whether representation geometry accounts for generalization performance, theory-derived error estimates were compared with empirical performance on the *SimplifiedRPM* task. Across all four models and 15 different held-out rule splits, the empirical error closely aligns with the theoretical curve $\varepsilon_a \approx H(\mathrm{SNR}_a)$ (Fig. 2.4b–e). This agreement relies on a second-order approximation in which each rule manifold is summarized by its centroid, dimensionality, and variability, leading to an approximately Gaussian decision variable. Despite neglecting higher-order or highly localized nonlinear structure, this approximation captures task-relevant variation in generalization performance across models with substantial architectural and representational differences.

That the same functional form holds across four architectures with different inductive biases, different depths, and different representational dimensionalities is the central validation of the framework. It means that the effect of architecture on relational generalization under this readout is captured, to the accuracy of the approximation, by these four measurable geometric quantities — and therefore that architectures can be compared on the geometry they produce rather than only on the errors they make.

The theory further identifies three mechanisms that facilitate generalization: increasing signal, expanding dimensionality D, and reducing signal–noise overlap. Across architectures, training consistently increases SNR but via distinct geometric routes (Fig. 2.4f–i). The SCL model primarily improves SNR through strong signal amplification, accompanied by reduced dimensionality and

pronounced overlap compression. In contrast, other architectures exhibit more balanced reorganization, combining moderate signal gains with increased dimensionality and reduced overlap. These results demonstrate that generalization arises from structured trade-offs among multiple geometric factors rather than from improvement along any single one. Applying the same decomposition to relation-level differences further reveals that variation in generalization difficulty across abstract relations is driven primarily by differences in centroid separation. Supplementary analyses rule out a uniform-geometry explanation for SCL's alignment with human difficulty ordering and instead identify rule-dependent signal heterogeneity as the source (Appendix A.9).

Table 2.3: Geometric quantities used in the rule-manifold analysis.

Symbol	Name	Description
$\mathbf{x}_i^a$	Relational representation	Representation of the ith row instantiating rule a.
N	Representation dimension	Dimensionality of the relational representation space.
$\mathcal{M}_a$	Rule manifold	Set of representations corresponding to rule a across different instantiations.
$\mathbf{x}_0^a$	Manifold centroid	Mean representation of rule a, $\mathbf{x}_0^a = \frac{1}{\lvert\mathcal{M}_a\rvert} \sum_{\mathbf{x}\in\mathcal{M}_a} \mathbf{x}$.
$\mathbf{u}_{a,k}$	Principal axis	kth principal direction of variability within the rule-a manifold.
$\lambda_{a,k}$	Axis variance	Variance along principal axis $\mathbf{u}_{a,k}$ (eigenvalue of Σ_a); $\sqrt{\lambda_{a,k}}$ is the radius along that axis.
R_a^2	Squared radius	Overall scale of the rule manifold, $R_a^2 = \operatorname{tr}\ \Sigma_a = \sum_{k=1}^{N} \lambda_{a,k}$.
D_a	Manifold dimensionality	Dimensionality of the rule manifold, $D_a = \frac{\left(\sum_{k=1}^{N} \lambda_{a,k}\right)^2}{\sum_{k=1}^{N} \lambda_{a,k}^2}$.
$\Delta\mathbf{x}_0$	Centroid separation	Normalized difference between rule centroids, $\Delta\mathbf{x}_0 = (\mathbf{x}_0^a - \mathbf{x}_0^b)/R_a$.
$\lVert\Delta\mathbf{x}_0\rVert^2$	Signal	Squared magnitude of centroid separation.
$\text{bias}_{a,b}$	Bias	Relative scale difference between rule manifolds, $\text{bias}_{a,b} = \frac{R_b^2}{R_a^2} - 1$.
$\mathbf{U}_a$	Scaled axes	Matrix of principal axes scaled by their radii in units of R_a, $\mathbf{U}_a = [\mathbf{u}_{a,1}\sqrt{\lambda_{a,1}}, \ldots, \mathbf{u}_{a,N}\sqrt{\lambda_{a,N}}]/R_a$.
$\lVert\Delta\mathbf{x}_0 \cdot \mathbf{U}_a\rVert^2$	Signal–noise overlap	Alignment between centroid separation and dominant variability directions.
$\text{SNR}_{a,b}$	Signal-to-noise ratio	Geometric quantity governing generalization error (Eq. 2.7).

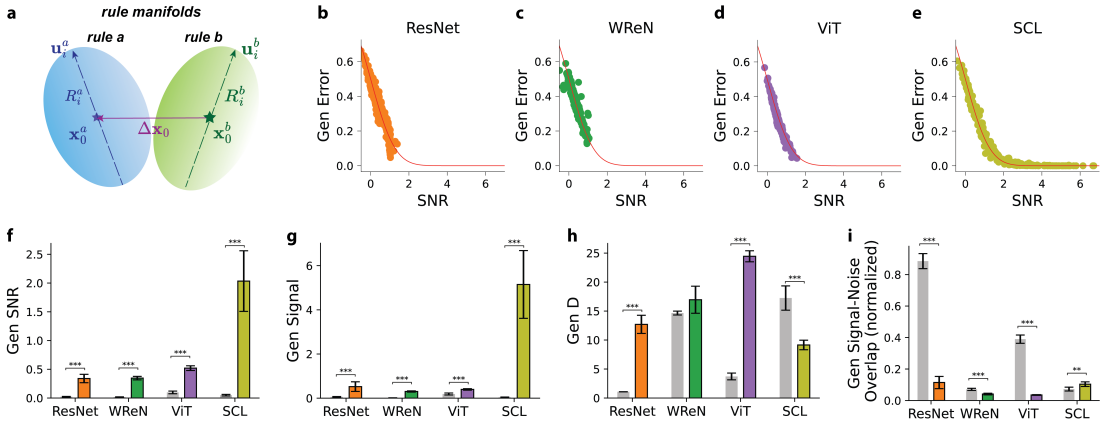


Figure 2.4: Generalization error is well described by the geometry of relational representations. (a) Illustration of rule manifolds in relational representation space. Each rule defines a manifold consisting of representations of all rows instantiating that rule. Two example manifolds are shown: rule a (blue) and rule b (green). Each manifold is summarized by its centroid ($\mathbf{x}_0$) and internal variability along principal directions ($\mathbf{u}_i$) with associated radii (R_i). Generalization between an ordered rule pair is governed by the centroid separation signal $\Delta\mathbf{x}_0$, together with manifold dimensionality and signal–noise overlap. (b–e) Empirical generalization error for held-out rule pairs as a function of the theory-predicted signal-to-noise ratio (SNR) for ResNet (b), WReN (c), ViT (d), and SCL (e). Each point corresponds to one ordered held-out rule pair, aggregated across 15 held-out rule splits (20 rule pairs per split), with generalization error evaluated using 500 *SimplifiedRPM* trials per rule pair. Solid red curves show the theoretical prediction $\varepsilon \approx H(\mathrm{SNR})$, where H is the Gaussian tail function. (f–i) Generalization SNR and its geometric components for relational representations on held-out rules. Grey bars denote untrained models at initialization, and coloured bars denote trained models. For each bar, SNR-related quantities are computed for 20 held-out rule pairs per split (5 held-out rules), with relational manifolds estimated from 1,000 test samples per rule, then averaged within each split and across 15 held-out rule splits. Bars show means across splits, and error bars indicate 95% confidence intervals. Training increases generalization SNR across all architectures, but via distinct geometric routes involving different combinations of signal amplification, dimensionality change, and signal–noise overlap reduction. Statistical significance was assessed using Wilcoxon signed-rank tests with false discovery rate correction; asterisks denote FDR-adjusted significance ($^{*}q < 0.05$, $^{**}q < 0.01$, $^{***}q < 0.001$).

2.6 Layer-wise geometry reveals model-specific generalization strategies

Different architectures exhibit distinct geometries in their relational representations, raising the question of how these geometries arise as networks deepen. The analyses in the previous section characterized the final relational representation, the layer immediately preceding the classification head. But that representation is the endpoint of a long sequence of transformations, and the question of *where* relational structure appears is as informative as the question of how good it eventually becomes. To address it, we extracted representations from every intermediate layer of each network, projected them to a fixed dimensionality to make widths comparable, and applied the full geometric decomposition at each stage.

2.6.1 SNR trajectories across depth

Layer-wise analysis of generalization SNR revealed strikingly different patterns of relational generalization across architectures (Fig. 2.5e–h). In ResNet-50 and ViT, SNR increases gradually across successive layers, consistent with a progressive refinement of relational representations (Fig. 2.5e,g). WReN reaches a plateau early, at the pairwise aggregation stage, with subsequent MLP layers providing little additional improvement, indicating an early bottleneck in relational abstraction (Fig. 2.5f). This is a direct consequence of its inductive bias: the pairwise relation module extracts what it is built to extract, and what it does not capture at that stage the downstream MLP cannot recover. The SCL model exhibits a markedly different trajectory: SNR rises sharply after the shared convolutional stage, at the first MLP layer (Fig. 2.5h), reflecting an architecture in which panel-wise features are computed first, and global relational structure is assembled abruptly at a later stage.

Together, these patterns show that relational structure does not emerge uniformly with depth but is instead localized to particular layers in an architecture-dependent manner, reflecting different strategies for relational generalization.

2.6.2 Decomposing the layer-wise trajectories

To identify the geometric mechanisms underlying these layer-wise differences, we decomposed the evolution of its constituent terms across network depth (Fig. 2.5i–p). While the signal generally increases with depth, its trajectory often diverges from that of SNR, indicating that signal amplification alone cannot account for relational abstraction. In ResNet-50, for example, SNR increases steadily across layers even though the signal remains nearly constant until a sharp rise at the final pooling stage (Fig. 2.5e, i). In SCL, the pattern is reversed in time: SNR improves earlier than the signal, while subsequent signal amplification fails to produce further gains in SNR (Fig. 2.5h,l). These divergences between signal and SNR demonstrate that signal amplification alone cannot account for generalization and point to a critical role for the other geometric factors.

Examining representational dimensionality — D, the rule-manifold participation ratio defined in Eq. 2.5 — reveals non-monotonic, architecture-specific expansion–compression dynamics across layers (Fig. 2.5m–p). In ResNet-50, dimensionality expands substantially in early layers before compressing at later stages (Fig. 2.5m), which explains why SNR improves well before the signal increases: the early gains are purchased by dimensional expansion, not by separating the rule centroids. By contrast, ViT maintains a consistently low dimensionality across most layers, with only a modest expansion in the final attention block (Fig. 2.5o), consistent with its gradual, signal-aligned improvement in SNR. In SCL, dimensionality increases through the convolutional layers, accounting for the early rise in SNR that precedes signal amplification, and the sharp compression of dimensionality in the final MLP layer (Fig. 2.5p) suppresses further gains in SNR despite continued signal growth.

Together, these results demonstrate that generalization is governed not by the signal alone but by the coordinated dynamics of signal and representational dimensionality. They also make a methodological point: a layer that improves relational separability need not be a layer at which anything rec-

ognizable as a "relational feature" appears, since the improvement may come entirely from a change in the shape of the variability rather than from a change in where the manifolds sit.

The robustness of these layer-wise trends was assessed across both training splits and random initializations. SCL, ResNet, and WReN show highly stable dimensionality trajectories (mean Spearman $\rho = 0.92$–0.99 across splits, $\rho \approx 0.94$–0.95 across seeds). In contrast, ViT is substantially less consistent ($\rho \approx 0.7$ and ≈ 0.74 respectively), consistent with its weakly varying dimensionality across intermediate attention layers. The late-layer dimensionality compression in SCL, which carries much of the interpretive weight above, was negative in all 15 splits. Full analyses, including two-dimensional PCA visualizations of held-out rule manifolds across layers, are given in Appendix A.3.5.

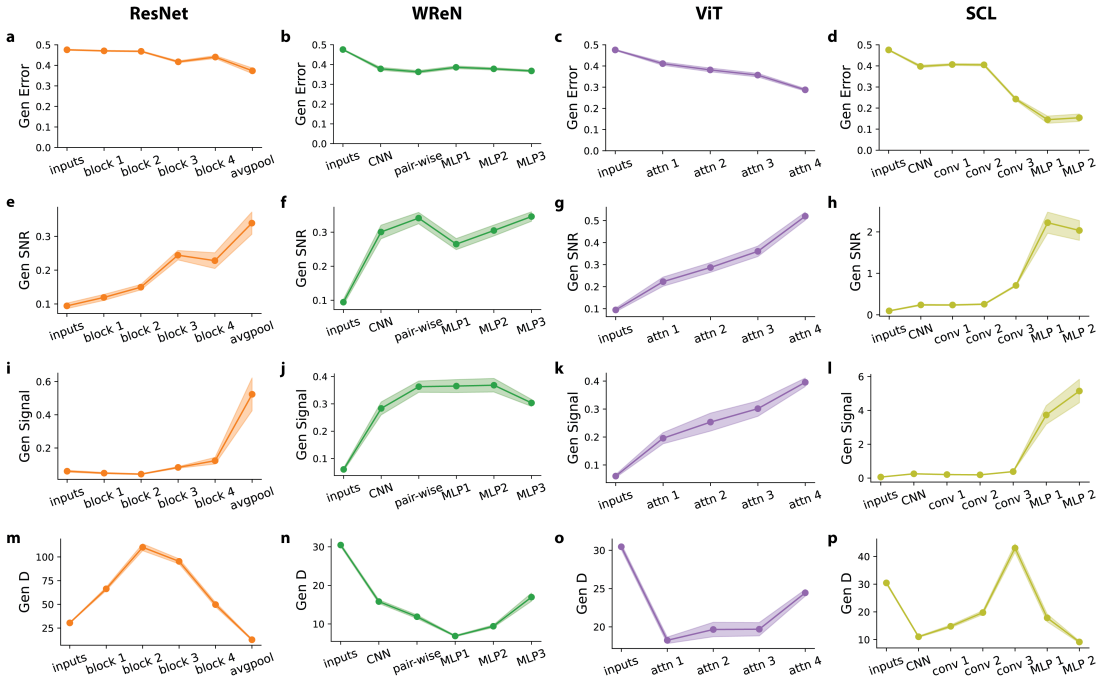


Figure 2.5: Layer-wise evolution of generalization geometry across architectures. Columns are ResNet-50 (a, e, i, m), WReN (b, f, j, n), ViT (c, g, k, o) and SCL (d, h, l, p); the horizontal axis of each panel lists that model's stages from the input to the layer before the classification head. (a–d) Generalization error on held-out rules across network layers for each model. Errors are averaged over 15 random held-out rule splits; shaded bands in every panel are s.e.m. across splits. For each layer, representations are randomly projected to a fixed dimensionality ($N = 400$) before computing errors and geometric quantities. Generalization error for each rule pair is evaluated using 500 *SimplifiedRPM* trials. (e–h) Signal-to-noise ratio (SNR) of Eq. 2.7 for held-out rule pairs. (i–l) Signal, $\|\Delta \mathbf{x}_0\|^2$. (m–p) Dimensionality ($D$) of the held-out rule manifolds. "Gen" in the axis labels denotes quantities measured on held-out rules.

2.7 Structured compression and the signal–dimensionality trade-off

Geometric theory predicts that both increased signal and high dimensionality should support generalization. Yet in the layer-wise analysis, model layers where the signal increases are typically layers where D decreases (Fig. 2.5i,m for ResNet-50; Fig. 2.5l,p for SCL). This inverse relationship is not specific to model layers: across all four architectures examined, signal and D exhibit a consistent anti-correlation throughout training (Fig. 2.6a–d).

The trade-off is not a mathematical necessity. Signal and D are geometrically independent prop-

erties: centroid separation between rule manifolds can increase without changing the dimensionality of the representation, and dimensionality can be reduced without directly affecting centroid distances. One can readily construct representations in which both are large. That they nevertheless move in opposite directions across architectures, across training, and — as Section 2.8 shows — across training objectives, demands an explanation. Why does this trade-off between generalization signal and D consistently emerge?

2.7.1 Held-out rules align with the training-rule subspace

To address this question, the structure of the learned rule representations was examined. Across architectures, representations of the training rules were found to lie in a low-dimensional subspace: the variance spectrum drops sharply beyond the top 35 principal components, matching the number of training rules. This in itself is unsurprising — a classifier trained to separate 35 categories has every reason to arrange them in a subspace of that rank.

What is not obvious is that representations of the five held-out rules also align with this same subspace, despite never having been seen during training. Specifically, the fraction of variance of the held-out rules explained by the top 35 principal components derived from the training rules increases systematically with the generalization signal across models (Fig. 2.6e–h). The subspace that training carves out for the rules it knows is progressively also the subspace that comes to contain the rules it does not.

2.7.2 The structured compression hypothesis

These results suggest that training induces a *structured compression* of the representation space. Rather than randomly collapsing representations, training progressively aligns unseen rule representations with a subspace defined by the training rules. Under this hypothesis, the observed

signal–D trade-off emerges as a by-product of the alignment process: representations become lower-dimensional precisely because they concentrate variance along rule-discriminative directions, and the same concentration that raises the signal removes the independent directions that dimensionality counts.

To evaluate this hypothesis directly, a polarization regularizer was added that encourages variance to concentrate along fewer directions in feature space while leaving the architecture and task objective unchanged (Appendix A.10). If the interpretation above is right, then imposing concentration should not merely reduce dimensionality but should also raise the generalization signal, since the two are meant to be two faces of the same reorganization. This is what happens: the regularizer reduced dimensionality. It was accompanied by increased generalization signal across held-out splits, while generalization error did not improve and was slightly higher than the baseline. The regularizer therefore reproduces the geometric signature of the trade-off, a higher signal bought with lower dimensionality, without reaching a better position along it. These observations are consistent with the structured compression hypothesis and suggest that representational alignment may contribute to the emergence of the generalization signal.

The trade-off is therefore not an obstacle that better training could remove, but a structural feature of the representations these networks form — which raises the question of whether an objective that is aware of both terms could position a model at a better point along it.

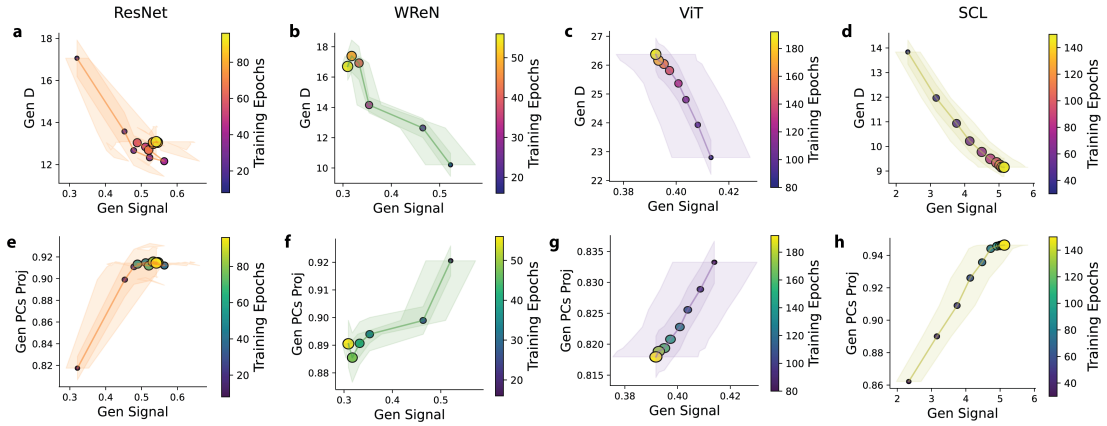


Figure 2.6: Structured compression underlies the trade-off between generalization signal and dimensionality. (a–d) Generalization signal versus dimensionality (D) across training for four architectures (ResNet, WReN, ViT, SCL). (e–h) Alignment of held-out rule manifolds with the low-dimensional subspace defined by training rules across training for all four architectures. Alignment is quantified as the variance in held-out rule representations explained by the top 35 principal components of training rule representations. For all panels, solid lines indicate averages across 15 random held-out rule splits, with shaded regions denoting s.e.m. across splits. Dot size and color encode training epoch, with later epochs shown as larger and brighter points. Generalization signal exhibits a consistent negative correlation with dimensionality (a–d) and a positive correlation with subspace alignment (e–h), quantified using Spearman's rank correlation; statistical significance was assessed using Stouffer's test.

2.8 Direct optimization of SNR improves generalization

Standard objectives such as cross-entropy do not explicitly optimize the geometric factors that theory links to generalization. They optimize separability of the *training* rules under a learned linear readout, and any improvement in the geometry of held-out rule manifolds is incidental. This raises the question of whether the geometry can instead guide training directly. Because Eq. 2.7 is differentiable in the network's representations, we can use it not only to analyze a trained model but also to train one. To test this possibility, the *SimplifiedRPM* benchmark serves as a controlled testbed for an SNR-based objective.

2.8.1 The SNR loss

The objective directly maximizes the signal-to-noise ratio between rule manifolds. In each mini-batch, m distinct relational rules are sampled uniformly without replacement from the training rules, and P rows are sampled for each, giving a batch of mP rows. All rows are passed through the network to obtain relational representations, and for each rule the manifold statistics required to evaluate Eq. 2.7 — centroid and covariance structure — are estimated from its P samples. The unit of optimization is therefore not an individual sample pair but the full set of P samples defining a rule-level manifold. The SNR is evaluated for all ordered rule pairs within the batch, yielding $m(m-1)$ rule-level comparisons per update; negatives are defined at the rule level rather than the instance level, all in-batch rule pairs are used, and no hard-negative mining is employed. The loss is

$$l_{\mathrm{SNR}} = \sum_{(a,b)} \exp\left(-\alpha \, \mathrm{SNR}_{a,b}\right), \tag{2.11}$$

where the sum is taken over all ordered rule pairs within the batch and α controls the sensitivity to SNR differences. The exponential form weights poorly separated rule pairs most heavily, so that optimization pressure concentrates where discriminability is worst. The loss used $\alpha = 0.5$, a batch composition of 32 rules $\times$ 16 samples, and a learning rate of 0.01, all selected by grid search on three held-out rule splits (Appendix A.11). Equation 2.11 applies uniform weighting across rule pairs, though the formulation readily extends to weighted variants that assign different importance to specific pairs (Appendix A.11).

2.8.2 SNR loss improves relational generalization

Training the SCL model with the SNR loss yields strong relational generalization, achieving an average generalization error of 0.096 ± 0.015 (mean $\pm$ 95% CI; Fig. 2.7a), substantially outperforming

cross-entropy training (0.153 ± 0.019). Optimization was stable across five random seeds, with final generalization error 0.129 ± 0.008 (mean $\pm$ s.d.) and no numerical instabilities or representational collapse.

To assess whether this advantage arises from explicitly optimizing the geometric separability predicted by the theory, rather than from more generic representation regularization, SNR loss was compared with two alternative geometry-based objectives: Prototypical loss[73], which encourages clustering around class centroids, and a supervised variant of variance–covariance regularization (VICReg)[74,75], which promotes high feature variance and low redundancy. These are informative comparisons because each optimizes a proper subset of what SNR loss optimizes: Prototypical loss targets centroid structure, VICReg targets the variance spectrum, and neither targets their relationship. SNR loss achieved the lowest generalization error across all objectives (Fig. 2.7a).

2.8.3 Every term of the objective is necessary

Ablation analyses clarify the necessity of the full formulation (Appendix A.11). Optimizing the signal alone increases centroid separation but induces dimensional collapse, degrading generalization — the failure mode that the signal–dimensionality trade-off of Section 2.7 predicts for any objective that pursues separation without regard to the variance spectrum. Conversely, removing the signal term abolishes centroid separation and leads to unstructured manifold expansion with near-zero SNR. Eliminating the overlap, bias, or dimensionality terms likewise impairs performance, indicating that each enforces a distinct geometric constraint. Together, these results show that robust relational generalization requires their coordinated regulation, uniquely enforced by the full SNR objective.

2.8.4 The trade-off persists across objectives

Because SNR loss is derived from the geometric framework, it optimizes the trade-off between signal and representational dimensionality rather than either in isolation, and the resulting representations show what that balance looks like. SNR-trained representations exhibit a modest reduction in signal relative to cross-entropy (Fig. 2.7b) but substantially higher dimensionality (Fig. 2.7c), and generalize better. The two alternative objectives sit on opposite sides: Prototypical loss increases dimensionality at the expense of a pronounced signal reduction, limiting generalization, whereas VICReg preserves higher signal but yields lower dimensionality.

Notably, across all objectives considered, increases in generalization signal are consistently accompanied by decreases in representational dimensionality (Fig. 2.7). The signal–dimensionality trade-off documented in Section 2.7 across architectures and across training therefore also holds across optimization objectives, including an objective explicitly constructed to be aware of both terms. What SNR loss achieves is not an escape from the trade-off but a better position along it.

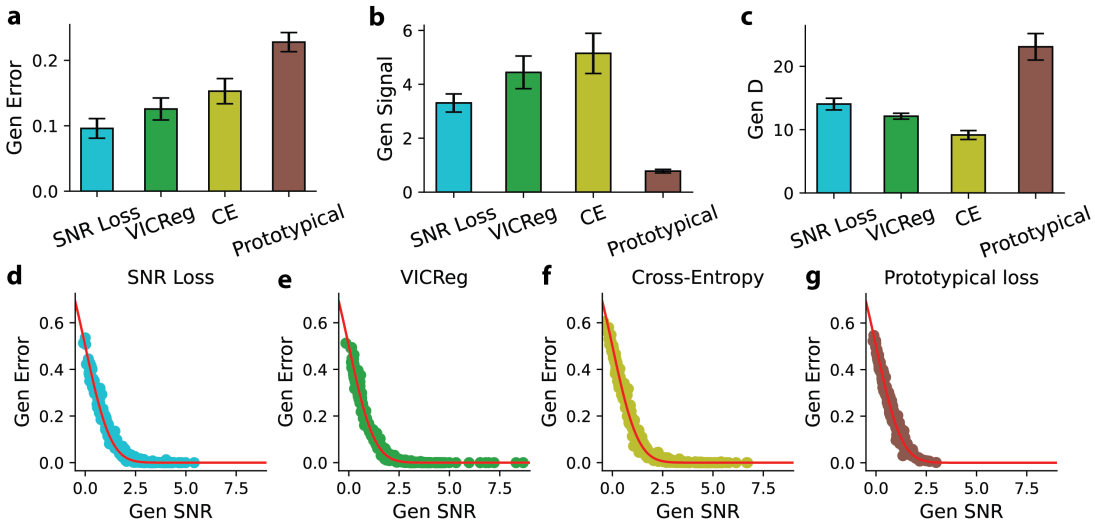


Figure 2.7: Direct optimization of SNR improves relational generalization. (a–c) Generalization error, signal strength, and dimensionality across training objectives for the SCL model. Bars denote the mean across 15 held-out rule splits, with error bars indicating 95% confidence intervals. Pairwise differences between loss functions were assessed using Wilcoxon signed-rank tests on split-level values. (d–g) Generalization error (y-axis) as a function of SNR (x-axis) for the SCL model trained with SNR loss (d), supervised VICReg (e), cross-entropy (CE) (f), and Prototypical loss (g). Each point denotes a held-out rule pair (20 pairs per split; 15 splits total). Red curves indicate the theory prediction $\varepsilon \approx H(\mathrm{SNR})$. p-values were corrected for multiple comparisons using the Benjamini–Hochberg false discovery rate procedure. All pairwise differences shown are statistically significant.

2.9 Generality of the geometric account

Everything established so far was established under the controls that define *SimplifiedRPM*: one relational rule per row, a fixed renderer, and a two-alternative decision. Those controls are what make the geometric analysis interpretable, but they also raise the obvious concern that the account holds only because the setting was engineered to make it hold. This section reports three stress tests, each relaxing one of the controls, and a fourth analysis identifying what distinguishes the architecture that generalizes best. Full details for all four are in the appendix; the purpose here is to establish the boundary of the claim.

2.9.1 Robustness to renderer shifts

The first concern is that the relationship between geometry and generalization might be tied to the particular visual statistics of the *SimplifiedRPM* renderer. To test this, zero-shot transfer was evaluated under renderer shifts that substantially alter low-level appearance while leaving relational structure untouched: sinusoidal texture modulation within object masks, global remapping of the fill palette relative to the background, and replacement of filled shapes by outlines of varying width. Each shift changes texture, contrast, or figure–ground statistics substantially, and none changes which relation a row instantiates. Across all renderer variants and five held-out rule splits per variant, the relationship between representation geometry and relational generalization persists (Appendix A.1.6). Relational generalization is therefore not an artifact of the specific rendering pipeline, and the geometric quantities remain predictive when visual appearance is perturbed.

2.9.2 Extension to the canonical RPM format

The second concern is more consequential: *SimplifiedRPM* deliberately removes the multi-row inference and multi-choice elimination that define the canonical task, and one might reasonably ask whether the framework survives their reintroduction. It does, with a modification that is itself informative.

In the canonical 3×3 format, the first two complete rows both instantiate the target rule, so a solver has two examples of the rule rather than one, and must select among eight candidate rows rather than two. The framework accommodates both changes directly. Two examples make the inference a two-shot rather than one-shot prototype problem, which rescales the noise terms of Eq. 2.7 by the number of shots m, yielding an m-shot prototype SNR (Appendix A.7); eight alternatives make a trial correct only if the target is preferred over all seven distractors, giving a trial error $\varepsilon_a = 1 - (1 - H(\mathrm{SNR}_{m=2}))^7$ for independent distractors.

Evaluating SCL zero-shot on this harder format — with no retraining, using models trained only on single-row *SimplifiedRPM* — gives a mean eight-choice completion error of 0.169 ± 0.018 (mean $\pm$ s.e.m.) against a chance error of 0.875, with a measured $\mathrm{SNR}_{m=2}$ of 3.02 ± 0.27. The measured SNR well describes the observed error under the modified decision rule. The geometric account therefore extends to a task format it was not fitted to, and the extension required changing only the decision rule, not the description of the representations.

2.9.3 Combinatorial rule composition

The third concern is the one-rule-per-row constraint, which guarantees that rule manifolds factorize cleanly. Real relational problems routinely involve several relations at once, and it is not obvious that a second-order manifold description survives the entanglement.

To test this, composite rules were constructed by applying two distinct held-out rules to different attributes of the same row — for example, constant shape combined with XOR color — and composites were paired that share one relational component while differing in the other, so that the resulting manifolds are partially entangled by construction. Using the same pretrained SCL models, evaluated zero-shot, composite rule discrimination across 30 ordered composite pairs from five splits yielded a mean SNR of 1.073 ± 0.204 and a mean generalization error of 0.237 ± 0.031, and the error remained well described by $\varepsilon \approx H(\mathrm{SNR})$ (Appendix A.8).

The scope of this result should be stated precisely. It shows that the geometric description survives *combinatorial attribute overlap* between rules, which is a genuine relaxation of the factorization assumption. It does not show that the framework handles hierarchical composition or interacting rules, in which the relations are not merely simultaneous but structurally dependent. That case remains open.

2.9.4 What distinguishes SCL's geometry

Finally, the human–model correspondence of Section 2.4 invites a deflationary explanation: perhaps SCL matches the human difficulty ordering simply because it imposes a more uniform geometry across rules, flattening idiosyncratic differences and leaving only the coarse relation-family structure that human data also show. This can be tested directly by measuring across-rule heterogeneity.

The deflationary account is wrong, and instructively so. Quantifying the coefficient of variation of rule-level SNR across held-out rules within each split, SCL shows *greater* across-rule heterogeneity than ViT (mean CV 0.612 versus 0.431; paired Wilcoxon signed-rank across splits, FDR-corrected $p = 0.0018$) and is indistinguishable from ResNet and WReN (all $p > 0.6$). The dissociation is sharp: ViT has the most uniform rule geometry of the four architectures and fails to reproduce the human ordering, whereas SCL retains rule-to-rule variation and reproduces it. Uniformity is therefore not what produces the correspondence.

Repeating the analysis component by component localizes the heterogeneity. It lies in the signal: SCL exhibits substantially greater across-rule variability in centroid separation than every other architecture (all $p < 3 \times 10^{-4}$). It does not lie in a uniform dimensional collapse — SCL preserves rule-dependent differences in dimensionality, exceeding WReN ($p = 0.0013$) and ViT ($p = 0.0040$) and matching ResNet ($p = 0.93$) — even though SCL does compress dimensionality at the level of individual layers (Section 2.6). What SCL does, then, is not to make all rules geometrically alike but to separate rule manifolds by amounts that differ systematically from rule to rule, and those differences are what align with human difficulty.

A layer-wise version of the same analysis locates where this alignment appears. At the input stage, generalization error is high and undifferentiated across relation families. Through the convolutional stages, the ordering is unstable, with logical relations initially easiest and arithmetic relations hardest. A stable progression < arithmetic < logical ordering — the human ordering — first emerges at the

first MLP layer and is preserved thereafter (Appendix A.9). This is the same layer at which SNR rises sharply in Section 2.6, which suggests that the emergence of relational separability and the emergence of human-like difficulty structure are the same event in this architecture rather than two independent properties of the trained model.

2.10 Discussion

A central challenge in cognitive science and artificial intelligence is to understand how neural systems generalize abstract relations beyond their training experience. This chapter takes a representational perspective, asking what structure emergent relational representations must possess to support generalization. To this end, the chapter introduced *SimplifiedRPM*, a controlled variant of Raven's Progressive Matrices that enables systematic evaluation of relational generalization in both humans and neural networks. Evaluation of multiple architectures — ResNet, WReN, ViT, and SCL — alongside human participants showed substantial variation in generalization to held-out relations, with SCL exhibiting the strongest performance and closest alignment with human difficulty ordering.

To understand these differences at the representational level, we applied a geometric framework originally developed for object classification [15] to relational reasoning. In this formulation, manifolds correspond to relational rule equivalence classes rather than object categories. The same SNR-derived geometric terms predict relational generalization performance, and directly optimizing SNR improves generalization. Extending the framework to rule manifolds further reveals a structured signal–dimensionality trade-off and architecture-specific patterns in the emergence of relational geometry across layers. Together, these results indicate that relational generalization is governed by a small set of interpretable geometric quantities: the separation between rule centroids (signal), representational dimensionality, and the overlap between signal and within-rule variability.

Crucially, these quantities evolve across network depth in architecture-specific ways. Vision Transformer exhibits gradual refinement of relational separation across layers, consistent with distributed integration through successive attention operations. In contrast, SCL shows a sharp transition in relational separability at the first MLP layer, reflecting its architectural design in which global relational structure emerges in later stages. The practical implication is that architectural comparison need not stop at aggregate accuracy: two models with similar generalization error can arrive there through entirely different geometric routes, and the decomposition makes those routes visible.

2.10.1 A neural hypothesis for relational difficulty

SCL reproduces the relative ordering of human relational difficulty across rule types, aligning both error patterns and response-time trends. Geometric analysis shows that easier relations are characterized by larger signals than more difficult ones, with centroid separation the principal source of relation-level differences (Section 2.5.5). These results suggest a concrete neural hypothesis: abstract relations may be encoded as rule-dependent manifolds in neural population activity, with behavioral difficulty reflecting systematic differences in their geometric organization. This hypothesis is testable with existing methods. The quantities in Eq. 2.7 are computable from population recordings, as they have been for object manifolds[14,15], and the prediction is specific: relations that humans find harder should correspond to rule manifolds with smaller centroid separation in the relevant population, not merely to noisier or weaker responses overall.

Although the models operate in a feedforward manner and human responses are slower and more variable, extended response times do not preclude a distributed relational substrate. In *SimplifiedRPM*, the relevant rule must be inferred for each new input rather than retrieved from prior experience, requiring participants to construct a task-specific decision boundary over internally generated relational representations. Additional time may therefore reflect iterative readout or refinement of these representations before decision-making, rather than a categorically different mech-

anism.

2.10.2 What the geometric account explains, and what it does not

It is worth separating the claims this chapter supports from those it does not, because a framework that predicts an outcome well can easily be over-read as explaining more than it does.

What the account explains is variation in relational generalization *given* a set of representations. Across four architectures, fifteen rule splits, every layer of every network, four training objectives, three renderer families, an eight-choice task format, and combinatorially composed rules, measured generalization error is well described by four geometric quantities computed from the representations. That is a substantial range of conditions for a four-parameter description with no free fitting, and it licenses treating the geometry as the quantity through which architecture and training determine performance under this readout: two models with the same rule-manifold geometry will generalize alike under the matching rule, whatever else differs between them. The geometry is measured on the same held-out representations whose error it predicts, so the claim is one of prediction from measured structure, not of an independent cause.

What the account does not explain is why a given architecture and training procedure produce one geometry rather than another. The framework is diagnostic, not generative. It shows that SCL generalizes well because large, rule-dependent amounts separate its rule manifolds, and that WReN plateaus early because its pairwise aggregation stage fixes the geometry that later layers inherit; it does not predict, before training, what geometry a proposed architecture will produce. Closing that gap would require a theory of how architectural constraints shape the learned covariance structure, which is a different and harder problem.

Nor does the account explain the signal–dimensionality trade-off itself. Section 2.7 documents it, shows that it is not a mathematical necessity, and identifies structured compression as the mechanism through which it arises — representations concentrate variance along rule-discriminative

directions, which raises the signal and removes independent directions in the same stroke. But why training should reliably take this route, when representations with both high signal and high dimensionality exist, is not answered here. The SNR loss shows only that an objective aware of both terms finds a better point along the trade-off, not one off it.

2.10.3 Relation to architectural and inference-time approaches

A wide range of models for relational reasoning incorporate architectural inductive biases, hierarchical grouping, or memory mechanisms to facilitate abstract rule learning[56,39]. More recent systems further enhance performance through multi-step inference-time computation, including predictive rollout search and tool-augmented reasoning[61]. These approaches improve generalization by allocating additional computation at test time.

In contrast, this study focuses on how relational structure is encoded within single-pass neural representations, before any explicit search or scaling. In this sense, the chapter distinguishes between the organization of representations and the inference procedures that operate over them. Search-based reasoning may compensate for imperfect representational geometry by allocating additional computation to resolve ambiguity. By isolating representational structure, the analysis identifies geometric principles that govern relational organization across architectures. Understanding how these principles interface with multi-step or search-based inference is an important next step, and the framework offers a natural way to pose it: one can ask how much test-time computation is required to recover a given amount of missing SNR.

Beyond differences in inference procedures, alternative frameworks posit fundamentally different representational forms. Symbolic and probabilistic accounts assume explicit rule representations or inference over structured hypothesis spaces[68,69], rather than distributed representations learned from sensory input, and the geometric framework does not directly apply to such models. Nevertheless, the shared dominant axes observed for trained and unseen rules suggest a connection to hier-

archical Bayesian theories that posit latent priors over relational hypotheses. From this perspective, distributed representations may implicitly encode a low-dimensional hypothesis space over relations, offering a bridge between geometric and probabilistic accounts of reasoning.

2.10.4 Limitations

One limitation of the representation-based approach is that generalizability is bounded by the statistics of the inputs the network has been exposed to during training. The framework describes the geometry that a representation has; it does not by itself say which training distributions will produce a good one. That said, the models are robust to modest shifts in the stimulus during reasoning: zero-shot transfer under renderer shifts that substantially alter texture, color palette, and object filling preserves the relationship between representation geometry and relational generalization (Appendix A.1.6).

SimplifiedRPM uses simple abstract objects and restricts each row to a single relational rule, thereby isolating abstract relational generalization from perceptual and combinatorial complexity. This is a deliberate choice — the controls are what make the geometric analysis interpretable — but it is also a genuine restriction on scope. As a controlled step toward greater structural complexity, the chapter showed that geometric predictors remain informative under increased task-format complexity, combinatorial rule composition, and renderer shifts (Appendices A.7–A.8). These extensions still preserve a single underlying relational rule and do not evaluate interacting rules or hierarchical composition. Real-world reasoning often involves multiple simultaneous relational structures as well as high-resolution visual inputs with fine-grained detail. Adaptive perception strategies, such as retrieval-augmented perception[76], address this challenge via coarse-to-fine selective crop retrieval and fusion, modifying how visual information is selected and combined before encoding. The present framework instead analyses the geometry of relational representations after encoding. Whether these geometric principles extend to adaptive high-resolution perception remains an open

question.

A second limitation concerns the approximation itself. The geometric analysis relies on second-order summaries that describe rule manifolds using centroids, dimensionality, and covariance structure. While these summaries capture relational separability in the controlled regime studied here, real-world representations may be multimodal, highly curved, or structured by context-dependent variability that global linear statistics cannot fully describe. In such settings, local geometry and nonlinear manifold structure may play a more prominent role. Whether the same geometric constraints extend to richer, naturalistic domains[77,78] therefore remains open.

A third limitation is that the human comparison, by construction, tests only the ordering of relational difficulty and not the mechanism producing it. Humans and models differ in learning history, in the number of exposures to each relation, and in whether responses are produced in a single pass. The correspondence reported in Section 2.4 constrains what a mechanistic account must explain; it does not supply one.

2.10.5 Outlook

In summary, the findings in this chapter advance the understanding of relational reasoning in neural networks using a geometric framework that links generalization performance to representation structure. By identifying geometric constraints that shape relational difficulty across both humans and models, this work offers testable hypotheses for future studies. It opens opportunities for extending geometric analysis to broader cognitive domains, including real-world visual reasoning and cross-modal linguistic tasks.

That extension is the subject of the next chapter. The quantities developed here are rule manifolds, centroid separation, participation-ratio dimensionality, and signal–noise overlap. The combined SNR is defined for any set of representations partitioned into equivalence classes, and its definition is not specific to vision or relational rules. Chapter 3 applies the same analytical vocabulary to

the internal representations of a speech model, asking whether the geometric organization that governs generalization in visual relational reasoning also governs it in a domain with entirely different inputs, architecture, and training objective.

2.11 Methods

2.11.1 *SimplifiedRPM* dataset

SimplifiedRPM is a rule-based visual reasoning dataset derived from Raven's Progressive Matrices. Each dataset instance consists of a single row of three panels governed by one abstract relational rule that specifies how an object's attribute values change across panels. The dataset comprises 40 relational rules, each defined by applying an abstract relation to a single object attribute. Relations span progression (constant, ± 1, ± 2), arithmetic (addition, subtraction), and logical (AND, OR, XOR) families and are applied to attributes including shape, size, shade, and number or position. Each row instantiates one and only one unambiguous relational rule, yielding a clean and interpretable relational structure. Section 2.3.1 describes attribute discretization, panel rendering, and the sampling procedure that enforces the one-rule-per-row constraint.

To ensure that performance on *SimplifiedRPM* cannot be explained by superficial visual cues, low-level image statistics were explicitly audited and feature-only baselines evaluated. These analyses indicate that low-level visual features neither encode relational rule identity nor support correct task performance. Appendix A.1 provides full dataset specifications, generation procedures, and validation analyses.

2.11.2 Model architectures

Four neural network architectures were evaluated. ResNet-50 follows the standard architecture [53], consisting of an initial convolutional layer, four groups of residual blocks, global average pooling,

and a fully connected classification head. WReN processes each panel with a shared convolutional block, constructs pairwise relational representations across panels, and aggregates them through a multilayer perceptron for classification. ViT embeds input patches into token representations and processes them through four stacked Transformer encoder blocks with multi-headed self-attention. SCL applies a shared convolutional block to each panel, followed by a hierarchy of one-dimensional convolutional layers operating across panels and a final multilayer perceptron; Fig. 2.8 shows an overview of the architecture and its panel-wise processing pipeline. Appendix A.2 tabulates complete layer-by-layer specifications for all four models.

Hyperparameters for WReN, ViT, and SCL were selected by grid search over architectural parameters, with model selection based on minimizing classification error on validation data from training rules. For WReN, this yielded a 2-layer, 512-dimensional relational module and a 3-layer, 256-dimensional classification MLP; for ViT, a 4-layer model with 12 attention heads and hidden dimension 96; for SCL, kernel size 1, channel sizes $\{64, 32, 5\}$, and an MLP hidden dimension of 400. ResNet-50 follows a standard architecture and did not require tuning. The full search grids are given in Appendix A.2.

2.11.3 Training and evaluation protocol

All models were trained and evaluated on the *SimplifiedRPM* dataset using a held-out rule generalization protocol. For each experiment, models were trained on 35 of the 40 relational rules and evaluated on the remaining 5 unseen rules. Training used supervised learning with a cross-entropy loss on one-hot rule labels, with 10,000 examples per training rule. Model selection was based on validation performance using 1,000 examples per rule, and test performance was evaluated on a separate set of 1,000 examples per rule.

Optimization used stochastic gradient descent with learning rate 0.1, momentum 0.9, weight decay 5×10^{-4}, and batch size 512. A learning-rate warmup was applied for the first two epochs,

followed by a plateau-based scheduler that reduced the learning rate by a factor of 0.2 when validation loss plateaued. Training proceeded until classification error on the validation set stabilized: 100 epochs for ResNet-50, 60 for WReN, 200 for ViT, and 150 for SCL. All models trained stably across five random seeds; we report computational costs in Appendix A.2.

Generalization to unseen rules was assessed using the *SimplifiedRPM* task. For each pair of held-out rules (20 ordered pairs per split), 500 unique trials were generated by pairing a sample row and a correct choice row from one rule with an incorrect choice row from the other. Model predictions were based on Euclidean distances between row representations, and a trial was counted as correct if the correct choice was closer to the sample than the incorrect one. Generalization error was computed as the fraction of incorrect trials, averaged across all rule pairs. All reported results are aggregated over 15 random training/held-out rule splits.

2.11.4 Extraction of relational representations

For all analyses of relational structure, representations were extracted from the final layer preceding the classification head: the output of the global average pooling layer for ResNet-50, the aggregated relational embedding following the relation module for WReN, the final Transformer encoder block for ViT, and the output of the second MLP layer immediately before the linear classifier for SCL.

For layer-wise geometric analyses, representations were extracted from each intermediate layer using 1,000 test examples per rule. To ensure comparability across layers and architectures with different widths, representations were randomly projected to a fixed dimensionality ($N = 400$) before computing geometric quantities. For each rule a and layer, we evaluated the dimensionality D_a as the participation ratio in Eq. 2.5.

2.11.5 Human behavioural experiment

Human performance on the *SimplifiedRPM* task was evaluated in $n = 25$ participants recruited via Amazon Mechanical Turk. We restricted participation to workers with a master's qualification and an approval rating above 95% to ensure data quality. All participants provided informed consent, and the study protocol was approved by the Institutional Review Board at Boston Children's Hospital, Harvard Medical School (IRB x08-10-0457). The study was conducted in accordance with the Declaration of Helsinki.

The task consisted of 130 *SimplifiedRPM* trials spanning 13 relational rule pairs (10 trials per pair). Each participant completed 60 trials randomly sampled from this pool to minimize sequential learning effects, and selected the correct answer from two alternatives on each trial. Participants were allowed unlimited time to respond and were compensated based on task accuracy. For each trial, both response accuracy and response time were recorded. All methods were performed in accordance with the relevant guidelines and regulations.

To minimize low-level perceptual difficulty, the human experiment was restricted to perceptually discriminable rule pairs, excluding rules based on subtle shade differences. Before the main experiment, participants completed three practice trials using rule pairs distinct from those tested, with feedback and explanations of the correct answers, to familiarize them with the task structure and response format. We ensured data quality using predefined participant- and trial-level exclusion criteria, including incomplete participation, failed attention checks, and implausible response times.

To enable direct comparison with behavior, the SCL model's generalization performance was evaluated on the same *SimplifiedRPM* trials used in the human experiment. For each of the 13 rule pairs, 10 SCL models were trained using different splits of the full rule set, ensuring that the target rule pair was always held out during training. Model generalization error was computed on the held-out rule pairs using the same trial set presented to human participants. Human error rates and

response times were then compared to model generalization error across rule pairs. Appendix A.5 provides additional details of the human experiment design, trial structure, and participant exclusion criteria.

2.11.6 Code and data availability

The code and dataset used in this study are publicly available at https://github.com/ShaneShang/SimplifiedRPM-Visual-Reasoning.

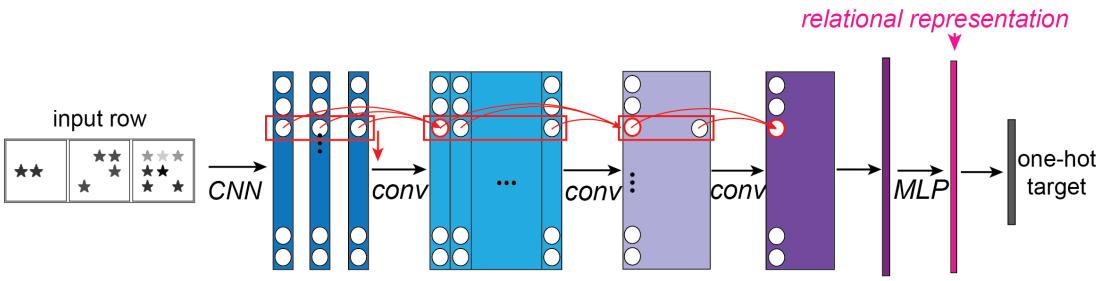


Figure 2.8: Overview of the Scattering Compositional Learner (SCL) architecture and its panel-wise processing pipeline. A shared convolutional block is applied to each panel, followed by a hierarchy of one-dimensional convolutional layers operating across panels and a final multilayer perceptron. The second MLP layer, immediately before the classification head, provides the relational representation used throughout the *SimplifiedRPM* analyses.

3

Geometric Signatures of Speech Recognition

This chapter describes work carried out in collaboration with Shailee Jain and Edward F. Chang (University of California, San Francisco). All analyses and text in this chapter are my own work; the collaboration contributed ongoing discussions of speech neuroscience and the design of the study. This work was presented at Cosyne 2025.

3.1 Summary of research in this chapter

Speech recognition requires mapping a continuous and variable acoustic signal onto discrete linguistic units. The acoustic realization of a phoneme or word varies with speaker, speaking rate, surrounding sounds, and sentence context, and linguistic boundaries are not marked by invariant acoustic events. Recognition must therefore distinguish categories across distributions of realizations rather than fixed templates. This chapter examines how successive stages of Whisper, a pretrained speech-recognition network, reorganize within-category variability relative to between-category differences, and where phonemes and words become geometrically separable, using the manifold framework developed in Chapter 2.

Whisper[18] is an encoder–decoder Transformer whose encoder transforms a log-mel spectrogram into acoustic representations and whose decoder generates the transcript token by token. Representations are recorded at 67 model stages, with the spectrogram as an input baseline. At each applicable stage, 200 utterances of each of 800 words define word manifolds, while 39 phoneme manifolds from TIMIT are followed through the encoder. We measure separability using the five-shot prototype readout from Chapter 2, and relate its error to the geometric ratio for every ordered pair of manifolds. Word manifolds are also compared between isolated and sentence-context conditions, using a position-matched control to distinguish the effects of preceding speech from those of input position. These analyses apply the geometric framework to a single model: the layerwise trajectories and context effects reported here are established for Whisper `large-v3`, and their generality across models remains to be tested.

Word separability follows a trajectory that a conventional linear probe eventually fails to resolve. Classification accuracy across all 800 words approaches the ceiling in the late encoder. It remains nearly constant through the late decoder, whereas five-shot error continues to vary over three orders of magnitude. Error reaches minima at encoder layer 27 and decoder layer 63, each before the out-

put of its respective stack, with an intervening plateau during which decoder separability remains near the spectrogram level. Across the stages examined, the geometric ratio closely predicts pairwise error, allowing this trajectory to be decomposed into its underlying geometric transformations.

The decomposition reveals two transformations that improve word separability. Across the first nine encoder layers, the signal returns to approximately its initial value, yet separability more than doubles as within-word variability spreads across more than three times as many dimensions and becomes less aligned with word-separating directions. From layer 9 to the encoder peak, improvement instead reflects a threefold increase in signal, with no sustained increase in dimensionality. Decoder improvement is likewise dominated by signal amplification rather than expansion. After each peak, signal loss and compression of variability reduce separability before the stack output.

The properties that distinguish word manifolds also change across depth. Duration difference predicts separability only at the entrance to each stack. In contrast, phoneme-sequence distance predicts it at both peaks and both outputs and is associated there with greater centroid separation. Semantic distance contributes little to the separability of isolated words but becomes increasingly predictive across the decoder when each word is preceded by its sentence. Thus, the geometry shifts from sensitivity to acoustic duration toward phonological structure, with semantic structure emerging only in context.

Phoneme manifolds follow a distinct trajectory. Their variability expands in the first encoder block without improving separability because the signal declines concurrently; subsequent improvement is driven entirely by signal amplification. Phoneme separability peaks at layer 18, nine layers before the word peak, where contrasts poorly separated in the spectrogram become resolvable. It then falls below the spectrogram level at the encoder output as within-phoneme variability becomes increasingly organized by word identity. Sentence context also affects the two stacks differently. Contextual and position-matched word manifolds remain similar through encoder layer 30, after which preceding speech introduces low-dimensional variability aligned with word-separating direc-

tions and causes separability to collapse. In the decoder, contextual variability is high-dimensional and only weakly aligned with those directions. Words therefore become more separable in sentence context than after silence, despite the weaker signal, because of the dimensionality and alignment of their variability.

Together, these results show that separability alone does not identify how a representation distinguishes linguistic categories: comparable performance can arise from changes in centroid separation, dimensionality, or alignment of within-category variability. The geometric decomposition distinguishes these mechanisms across processing depth, linguistic units, and contextual conditions, and yields quantities that can in principle be estimated from neural population recordings. The chapter uses these results to formulate predictions for the cortical speech hierarchy, delimit the conclusions supported by Whisper, and identify experiments that could test the remaining interpretations.

3.2 Background

3.2.1 From acoustic variability to linguistic units

Speech recognition requires mapping a continuous acoustic signal onto discrete linguistic units such as phonemes and words. The same word varies across speakers in fundamental frequency, formant frequencies, and voice quality[79]; changes in speaking rate alter segment durations nonuniformly[80]; and pronunciation varies with dialect, prosodic position, and reduction in fluent speech. Recognition must therefore identify the same linguistic unit across substantially different acoustic forms.

The temporal continuity of speech compounds this variability. Because the articulators move continuously from one sound to the next, the acoustic form of each segment depends on its neighbors, and cues to one phoneme extend into adjacent intervals; the formant transitions of a vowel, for example, carry information about the preceding consonant[1]. Connected speech also contains no invariant acoustic markers of phoneme or word boundaries. Listeners must therefore infer both the

identities and boundaries of linguistic units by integrating multiple acoustic and linguistic cues[81].

These problems unfold at different temporal scales for phonemes and words. A phoneme typically spans tens of milliseconds, although evidence about its identity can extend beyond its annotated boundaries. A word unfolds over several hundred milliseconds, and its recognition depends not only on the identity and order of its phonemes but also on duration, reduction, segmentation, and sentence context[82,83]. Indeed, at the same signal-to-noise level, words are recognized substantially more accurately in sentences than in lists[84]. Word recognition therefore cannot be reduced to a sequence of independent phoneme decisions; acoustic features, phonemes, words, and sentence context provide related but distinct descriptions of the same signal.

These properties motivate a geometric description of speech representations. Because each linguistic unit has many acoustically distinct realizations, it is represented not by a single template but by a distribution in representation space. Recognition depends on the organization of these distributions: categories become more distinguishable as the differences between them increase relative to their internal variability. In an end-to-end speech recognizer, both quantities may change across successive stages of computation. This chapter asks how the model reorganizes within-category variability and between-category differences across its hierarchy, and where phonemes and words become geometrically separable.

3.2.2 Internal representations in deep speech models

Modern speech-recognition systems are not explicitly supplied with phonemes, syllables, or words as intermediate representations; any organization corresponding to these units must instead emerge through training. Self-supervised encoders such as wav2vec 2.0[85] and HuBERT[86] learn from untranscribed speech by predicting masked portions of the input, whereas supervised or weakly supervised encoder–decoder models such as Whisper[18] learn to generate text from speech. Their successive layers provide a computational hierarchy in which to ask whether and where phonetic and

lexical structure emerges. This chapter examines that hierarchy in Whisper.

Layer-wise analyses suggest a broad progression from acoustic to phonetic and lexical information, although its trajectory depends on the model and training objective. In an end-to-end recognizer trained to generate character sequences, phoneme labels were most accurately decoded from lower layers and became less accessible toward the output[87]. In pretrained wav2vec 2.0, similarity to the acoustic input is greatest in early layers, phonetic information peaks in intermediate layers, and word identity and meaning peak later; the upper layers then return toward the acoustic input, a reversal removed by fine-tuning for transcription[88]. Across self-supervised speech models, word-level information is likewise concentrated in intermediate layers, but the location of its peak varies with the pretraining objective and model size[89]. Thus, depth orders linguistic information, but not according to a universal or necessarily monotonic trajectory.

Depth also changes how linguistic categories are organized, not merely how accurately their labels can be decoded. In a convolutional word classifier and a recurrent speech recognizer, word representations became progressively more separable across layers while speaker identity became less separable. The improvement in word separability arose because the realizations of each word became more tightly clustered and varied along fewer directions[90]. This work demonstrates that successive layers can reshape category geometry, but it examined architectures with a single processing stack rather than a complete encoder–decoder hierarchy.

Context provides another source of representational variability. In BERT, ELMo, and GPT-2, representations of the same word in different sentences become progressively less similar across layers, and a single static vector explains, on average, less than five percent of a word's variance at any layer[91]. Despite this contextualization, manifolds defined by syntactic categories, parts of speech, and combinatory categorial grammar tags become more separable with depth through reductions in their radius, dimensionality, and correlations[92]. Whether sentence context similarly reorganizes the geometry of word representations in speech models remains unknown: previous layer-wise analyses

examined words in sentences throughout and therefore did not compare the same words in isolation and in context[88,89].

This layer-wise organization is also relevant to the cortical hierarchy for speech. Processing progresses from spectrotemporal analysis in the primary auditory cortex to phonetic-feature selectivity in the superior temporal gyrus[8] and word-form recognition in the superior temporal lobe[10]. Correspondingly, early layers of trained speech models tend to predict primary auditory responses most accurately, whereas deeper layers better predict responses in the superior temporal gyrus and sulcus[17,93,94,95,96]. In Whisper specifically, encoder representations best predict intracranial responses in the superior temporal gyrus, while decoder representations best predict inferior frontal and higher-order language regions during natural conversation[97]. These correspondences do not imply that model layers reproduce cortical anatomy, but they indicate that depth indexes transformations relevant to biological speech processing.

Much of this evidence comes from probing, in which a classifier is trained to recover a label from each layer and its accuracy is treated as a measure of that label's accessibility. Probe accuracy depends jointly on the representation and the classifier, however, and a sufficiently flexible probe may learn distinctions only weakly expressed in its input[98]. Accuracy is also bounded and therefore ceases to distinguish layers once performance approaches the ceiling. More fundamentally, it does not reveal how categories are organized: the same improvement may arise from greater separation between category centers, reduced within-category variability, variability distributed across more or fewer dimensions, or a change in its alignment with the category-separating direction. Cortical encoding models have an analogous limitation because prediction accuracy measures correspondence between model and neural responses without specifying the category geometry of either space. These limitations motivate an analysis that directly decomposes representational separability.

How linguistic geometry changes across a complete encoder–decoder speech model, and how sentence context alters that geometry, therefore remain unresolved. This chapter addresses both

questions by tracing phoneme and word manifolds through Whisper's encoder and decoder and by comparing the same words in isolation and after their preceding sentence. The analysis separates changes in centroid signal from changes in the magnitude, dimensionality, and alignment of within-category variability, revealing transformations that probe accuracy alone cannot identify.

3.2.3 Whisper as an encoder–decoder model

Whisper[18] is a family of sequence-to-sequence speech models trained on approximately 680,000 hours of audio–text pairs collected from the web. This large and heterogeneous corpus spans multiple languages, speakers, recording conditions, and acoustic environments. Training is formulated as multitask conditional sequence generation: special control tokens specify the spoken language, whether the model should transcribe or translate into English, and whether timestamps should be emitted, while a no-speech token supports speech-activity prediction. Whisper's internal representations are therefore shaped by an end-to-end objective that maps acoustically variable speech onto structured sequences of linguistic tokens.

The model follows an encoder–decoder Transformer architecture (Section 3.3; Fig. 3.1). A 30-s audio segment is converted into a log-mel spectrogram and passed through a two-layer convolutional stem that projects the input into the model dimension and reduces its temporal resolution. After sinusoidal positional embeddings are added, a stack of Transformer encoder blocks uses bidirectional self-attention to produce a temporally resolved representation of the full audio segment. The autoregressive decoder receives previously generated text tokens together with learned positional embeddings. Each decoder block applies causal self-attention over the preceding token sequence, cross-attention to the encoder output, and a feedforward transformation. The final decoder state at each position is projected onto the vocabulary to predict the next token.

The encoder and decoder thus organize information in distinct representational domains. Encoder states are indexed by acoustic time and derived from the input signal, although bidirectional

self-attention allows each state to incorporate information from the full audio segment. These states remain temporally anchored even as they encode phonetic and lexical structure. Decoder states, by contrast, are indexed by positions in the output sequence and conditioned jointly on the encoded audio, the task-defining control tokens, and the preceding transcript. They therefore integrate acoustic evidence with an evolving linguistic context. The encoder–decoder boundary is not simply a transition between adjacent layers in a homogeneous stack; it marks a shift from frame-based representations of the acoustic signal to token-based representations constructed for autoregressive sequence generation.

This organization makes Whisper well suited for tracing how speech representations are reorganized over the course of recognition. Following phoneme and word representations from the acoustic input through the encoder and decoder makes it possible to distinguish geometric transformations associated with acoustic encoding from those associated with contextualized transcript generation. Whisper thereby provides a single end-to-end system in which to examine the transformation of continuous speech into discrete linguistic output.

3.2.4 Applying the geometric framework to speech

The geometric framework used in this chapter is developed in Section 2.5 and is not re-derived here; notation follows Table 2.3. At each processing stage, the responses evoked by all utterances of a word or phoneme define that category's manifold. Each manifold is characterized by its centroid, radius, principal axes, and effective dimensionality. These quantities describe, respectively, the mean response, the dispersion around that mean, the dominant directions of variation, and the number of directions over which the variation is distributed—the participation ratio is given in Eq. 2.5.

Separability is evaluated using a few-shot prototype readout. A small number of reference utterances from each of the two categories are averaged to form one prototype per category, and a test utterance is assigned to the category with the nearer prototype. We predict the expected error from

the signal-to-noise ratio in Eqs. 2.8 and 2.7, decomposed into the signal, bias, dimensionality, and signal–noise overlap terms introduced in Section 2.5. Appendix B.3 describes the estimator used in this chapter. We apply this decomposition throughout Whisper, from the input spectrogram to the final decoder block. Estimating each contribution separately lets us determine why separability changes between successive stages.

The within-manifold variation arises from natural differences among utterances, including variation in speaker, speaking rate, neighboring sounds, and sentence context. The geometric quantities characterize how each processing stage organizes this variability relative to category identity. Because Whisper is trained for sequence transcription rather than prototype classification, the measured separability is an emergent property of its representations rather than an objective that the model directly optimizes.

The same framework is used to analyze phoneme and word manifolds. At the encoder stage, both are measured within the same representational spaces, allowing direct comparison of their geometry. Word manifolds are also followed through the decoder, where representations are indexed by output tokens rather than acoustic time. Phoneme manifolds collect responses to short acoustic intervals, whereas word manifolds span multiple phonemes and vary additionally with sentence context.

Questions addressed in this chapter

This chapter addresses five questions. First, at which stages of Whisper do word representations become separable under the few-shot readout, and does the signal-to-noise ratio in Eq. 2.7 predict the measured error at each stage? Second, which geometric terms account for changes in separability across successive stages? Third, how do differences in acoustic form, phonemic content, and semantic meaning affect the separability of two words, and at what depths do these effects emerge? Fourth, does Whisper construct separable representations of phonemes and words through the same geometric transformations or through distinct ones? Fifth, how does the preceding sentence context

alter the representation of a word, and do these effects arise in the encoder, the decoder, or both? By tracing these properties across the encoder–decoder pathway, this chapter identifies where phoneme and word identity become separable and the geometric transformations through which this separation emerges.

3.3 Extracting word representations from Whisper

Whisper [18] is an encoder–decoder Transformer trained to transcribe speech. The encoder receives a 30-s log-mel spectrogram and passes it through a two-layer convolutional stem, whose second layer has stride 2 and reduces the temporal resolution to one frame per 20 ms. Sinusoidal positional embeddings are added to the resulting sequence of 1,500 frames, which is then processed by 32 pre-layer-normalized residual Transformer blocks with a hidden dimension of 1,280. Each block contains a self-attention sublayer and a multilayer perceptron, and the encoder output undergoes a final layer normalization (Fig. 3.1). The decoder generates text autoregressively using 32 Transformer blocks, each containing masked self-attention over preceding tokens, cross-attention over the encoder output, and a multilayer perceptron. Following a final layer normalization, the decoder states are projected onto the token vocabulary to predict the next token. Decoding is conditioned on a multitask prefix specifying the start of the transcript, language, and task. All analyses use the `large-v3` checkpoint in evaluation mode and at 32-bit floating-point precision. Additional architectural and training details are provided in Section 3.2.3, and parameters specific to representation extraction are reported in Section 3.10. Each analyzed encoder and decoder stage therefore provides a 1,280-dimensional population representation to which the geometric framework introduced in Chapter 2 applies consistently across model depth.

The stimuli comprise words spoken in read English sentences from LibriSpeech [100]. From the `test-clean` and `train-clean-360` partitions, 800 word types were selected, each represented by

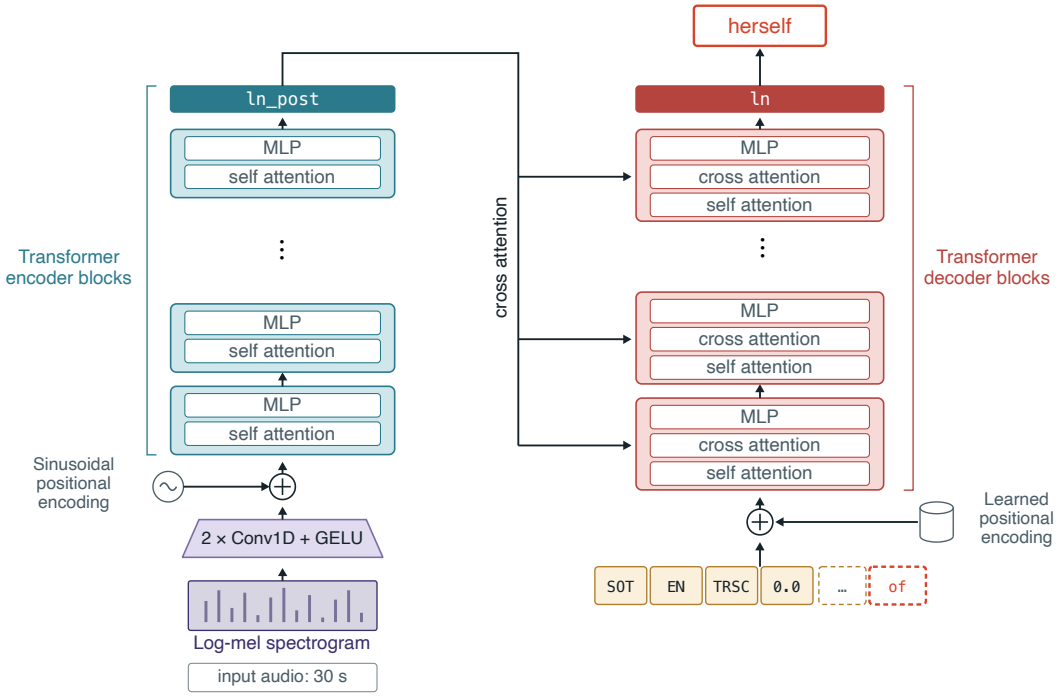


Figure 3.1: Architecture of Whisper `large-v3` and extraction of word representations. Audio is resampled to $16\,\mathrm{kHz}$ and converted into a 128-bin log-mel spectrogram using 25-ms windows with a 10-ms stride, yielding $3{,}000$ frames for a 30-s input. The encoder (teal) applies two convolutional layers with kernel size 3 and GELU activations; the second has stride 2, reducing the sequence to $1{,}500$ frames at 20-ms intervals. Sinusoidal positional embeddings are then added, followed by 32 pre-normalization Transformer blocks of width $1{,}280$ and a final layer normalization. The decoder (red) has the same width and depth but uses learned positional embeddings. Each block applies masked self-attention, cross-attention to the encoder output, and a multilayer perceptron. A final layer normalization and the tied token-embedding matrix project decoder states onto the vocabulary. The decoder prefix contains the start-of-transcript token (`SOT`), English-language token (`EN`), transcription-task token (`TRSC`), and initial timestamp token (`0.0`). In the in-sentence condition, the ground-truth transcript preceding the target word is appended to this prefix (dashed) as part of the current transcript rather than through Whisper's preceding-context mechanism. The state at the final input position predicts the first token of the target word, illustrated for *herself*. At each encoder stage, we obtain a word representation by averaging the residual stream over its time-aligned frames; at each decoder stage, we use the state at the final input position. Architecture adapted from Radford et al.[18]; `large-v3` uses 128 rather than 80 mel bins[99].

200 occurrences drawn from distinct sentences, yielding 160,000 word tokens in total. No model is trained or fine-tuned here. Word boundaries come from published forced alignments of LibriSpeech [101], produced with the Montreal Forced Aligner [102] rather than by Whisper, so the segmentation of the stimuli is independent of the model under study. Every occurrence is preceded by several seconds of speech in its sentence, and the audio from sentence onset to word offset fits the model's input window, so that each word can be presented in the in-sentence condition described below. Three exclusions applied at selection remove words that Whisper's vocabulary cannot produce as a single token, function words, and a manually specified list of numerals, proper names, and closed-class words (Section 3.10.3). The selection rules and the counts they leave are given in Appendix B.1.1.

Each utterance is presented to the model in one of two conditions. In the *single-word* condition, we excise the audio between the word's boundaries from its sentence, place it at the start of the input window, and zero-pad to 30 seconds. Hence, the model hears only the word. In the *in-sentence* condition, the input is the original sentence audio from its onset up to the end of the target word, again padded to 30 seconds. The model then hears everything that preceded the word, but nothing that followed it, and the word itself is acoustically identical in the two conditions: the same waveform, the same speaker, the same coarticulation with the preceding word. Differences in its representation can therefore arise from the preceding acoustic and linguistic context, and also from the word's position within the input window: the isolated word begins at the start of the window, whereas the in-sentence word begins wherever it falls in the sentence, so the two conditions also differ in the sinusoidal positional encoding added to the word's frames and in whether the word's onset is preceded by silence or by continuous speech. A position-matched control, in which the isolated word is preceded by silence so that it begins on the same frame as in its sentence, separates the word's position in the audio from the preceding speech in both stacks (Section 3.10.4); the Discussion returns to this point (Section 3.9.5). The in-sentence condition is used only in Section 3.8; all

other results are for isolated words.

Encoder representations follow a single rule at every depth. At the output of the convolutional stem, after each of the 32 encoder blocks, and after the final layer normalization, the residual stream, a 1500×1280 array over frames and units, is averaged over the frames that fall within the word's boundaries. A word lasting 300 ms contributes about 15 frames; we average over all of them, however many there are, so each utterance yields one 1280-dimensional vector per layer regardless of duration. This temporal pooling makes utterances of different lengths comparable in a fixed-dimensional space, while speaking rate remains one source of variability within a word's manifold. Pooling also discards the order of the frames.

Decoder representations are read at the position from which the model produces the word: the last input position, whose state predicts the word's first token. There, the residual stream is taken at the output of the token embedding, after each of the 32 decoder blocks, and after the final layer normalization. In the single-word condition, the decoder input is the task prefix alone. In the in-sentence condition, the true preceding words of the sentence follow the prefix as tokens.

These readouts define a single layer axis running from 0 to 66. On the encoder, index 0 is the output of the convolutional stem, before any block, and indices 1 through 32 are the residual stream after encoder blocks 1 through 32, so that the index counts the blocks applied; index 33 is the output of the encoder's final layer normalization, which is what the decoder attends to. On the decoder, indices 34 through 65 are the residual stream after decoder blocks 1 through 32, and index 66 is the output of the decoder's final layer normalization, from which the next token is read. The decoder input, the token embedding before any block, is not assigned an index: in the single-word condition it is the embedding of the same task prefix for every word and therefore carries no word information. Throughout the chapter "layer k" refers to this index, and figures label their ticks by stack so that locating a point on the axis requires no arithmetic. These representations define the layer-specific word manifolds whose separability and geometry are examined in the following sections.

3.4 Word separability across Whisper's depth

At each layer, the 200 utterances of a word define its *word manifold*, and the 800 words define 800 manifolds in the same 1280-dimensional space. The geometric description of each manifold and of each ordered pair of manifolds is exactly that of Section 2.5: centroid, radii, participation-ratio dimensionality D, signal, bias, and signal–noise overlap, combined into the signal-to-noise ratio of Eq. 2.7 and the predicted error $\varepsilon = H(\mathrm{SNR})$ of Eq. 2.8. Two conventions differ from Chapter 2 and are fixed here. First, because Whisper's width is the same at every layer, we apply no random projection; all quantities are computed in the full 1280-dimensional space. Second, the readout is the m-shot prototype classifier of Eq. A.4 with $m = 5$ (Appendix B.3): for an ordered pair of words, five sampled reference utterances from each word are averaged to form two prototypes, a separate held-out utterance is assigned to the nearer prototype, and the SNR that predicts the resulting error is $\mathrm{SNR}_{m=5}$. We use $m = 5$ for every curve in the chapter so that we can compare separability across datasets, for different features, and in the two context conditions with the same readout. Unless stated otherwise, every quantity plotted against layer is a mean over all 800×799 ordered word pairs.

3.4.1 Linear decodability of word identity across depth

To assess how the linear accessibility of word identity changes across Whisper's depth, we train a linear classifier at each layer to predict word identity from the corresponding representation. This provides a conventional measure of separability against which to compare the few-shot readout of Section 3.4.2. The classifier is trained jointly on all 800 words, making this an 800-way classification task with chance accuracy $1/800$, and is evaluated by five-fold cross-validation over the 200 occurrences of each word (Section 3.10.6). Figure 3.2 shows accuracy across the layer axis defined in Section 3.3. Word identity is already linearly decodable at the output of the convolutional stem,

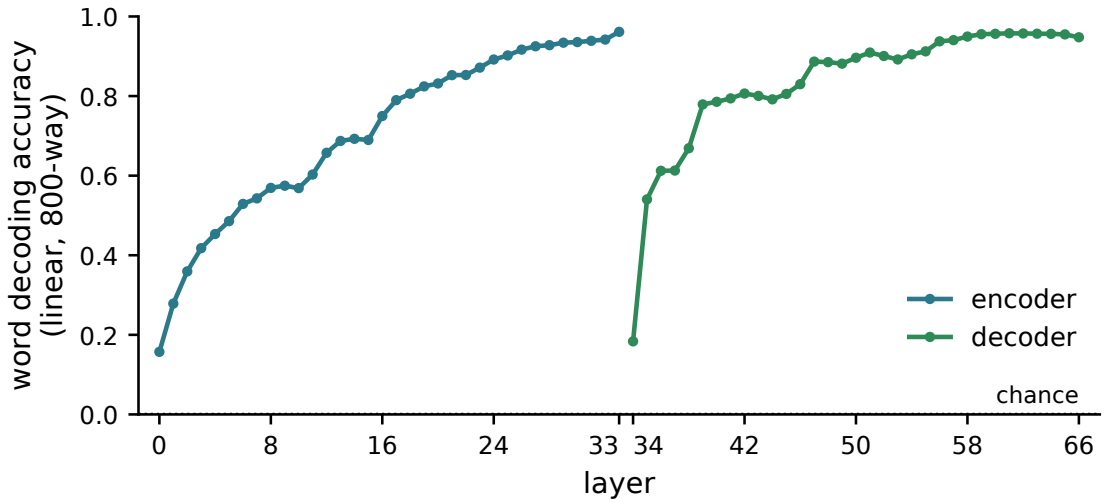


Figure 3.2: Linear decodability of word identity across Whisper's depth, single-word condition. Accuracy of an 800-way linear classifier trained on the representation at each layer and evaluated by five-fold cross-validation over the 200 occurrences of each word; encoder in teal (layers 0–33), decoder in green (layers 34–66). Lines are the mean over folds and bands $\pm$ one s.e.m. over folds. Chance is $1/800$.

before any Transformer block, where accuracy is 0.16. Accuracy then rises through the encoder, apart from two dips of less than 0.01 at layers 10 and 15, passing 0.5 at layer 6 and 0.9 at layer 25 before reaching 0.96 at the encoder output. The decoder input carries no word information in the single-word condition, since every input contains the same task-prefix token at the same position. Once word-specific information becomes available through cross-attention to the encoder output, accuracy is 0.18 at the first decoder layer, rises to 0.78 by layer 39 and reaches 0.94 by layer 57, after which it remains between 0.94 and 0.96 through the final layer.

With 800 classes, linear accuracy does not saturate within a few layers. Still, it continues to improve through most of each stack, showing that word identity is linearly decodable from the encoder's early layers onward. It does not say how the representation achieves this.

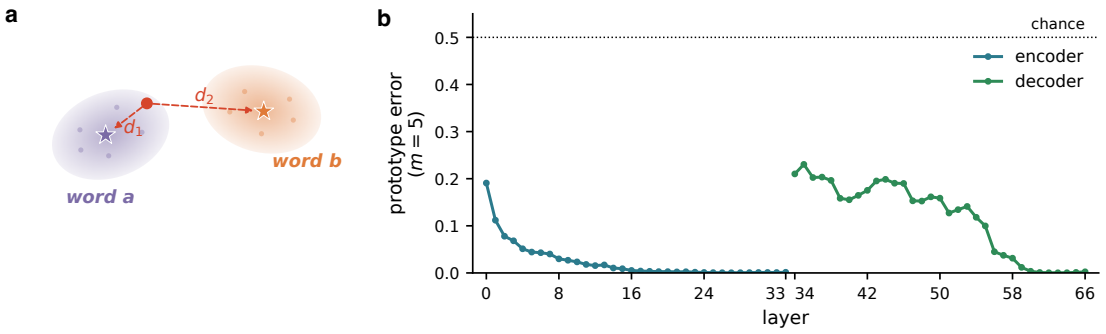


Figure 3.3: Few-shot prototype readout of word identity in the single-word condition. (a) For one ordered word pair at one layer, five reference occurrences of each word (faded points) are averaged to form its prototype (star). A held-out occurrence of word a (red) is classified by its distance from the two prototypes. Pairwise error is $P(d_2 \leq d_1)$ over held-out occurrences and reference draws. (b) Mean error across all 800×799 ordered pairs and model stages. Encoder stages are shown in teal and decoder stages in green; bands denote $\pm$ one s.e.m. across words. Chance error is 0.5.

3.4.2 A few-shot readout reveals structure not resolved by linear accuracy

We next measure word separability using the five-shot prototype readout introduced in Chapter 2 (Fig. 3.3a). For each ordered pair of words, five occurrences of each word are averaged to form two prototypes, and a held-out occurrence of the first word is assigned to the nearer prototype. The readout requires no fitted weights and is evaluated independently for all 800×799 ordered pairs.

Mean prototype error follows a strongly nonmonotonic trajectory across depth (Fig. 3.3b). It decreases from 0.22 at the log-mel spectrogram and 0.19 after the convolutional stem to 0.04 at encoder layer 6. It reaches a minimum of 4×10^{-4} at layer 27, then rises to 1.4×10^{-3} at the encoder output.

At the transition to the decoder, error returns to 0.21 and remains between 0.12 and 0.23 through layer 54. It then falls sharply to a second minimum of 3×10^{-4} at layer 63 before rising to 2.4×10^{-3} at the final layer. Thus, although linear accuracy improves through most of both stacks, the prototype readout reveals distinct phases of increasing and decreasing separability. The next subsection tests whether the geometric theory of Chapter 2 predicts this trajectory.

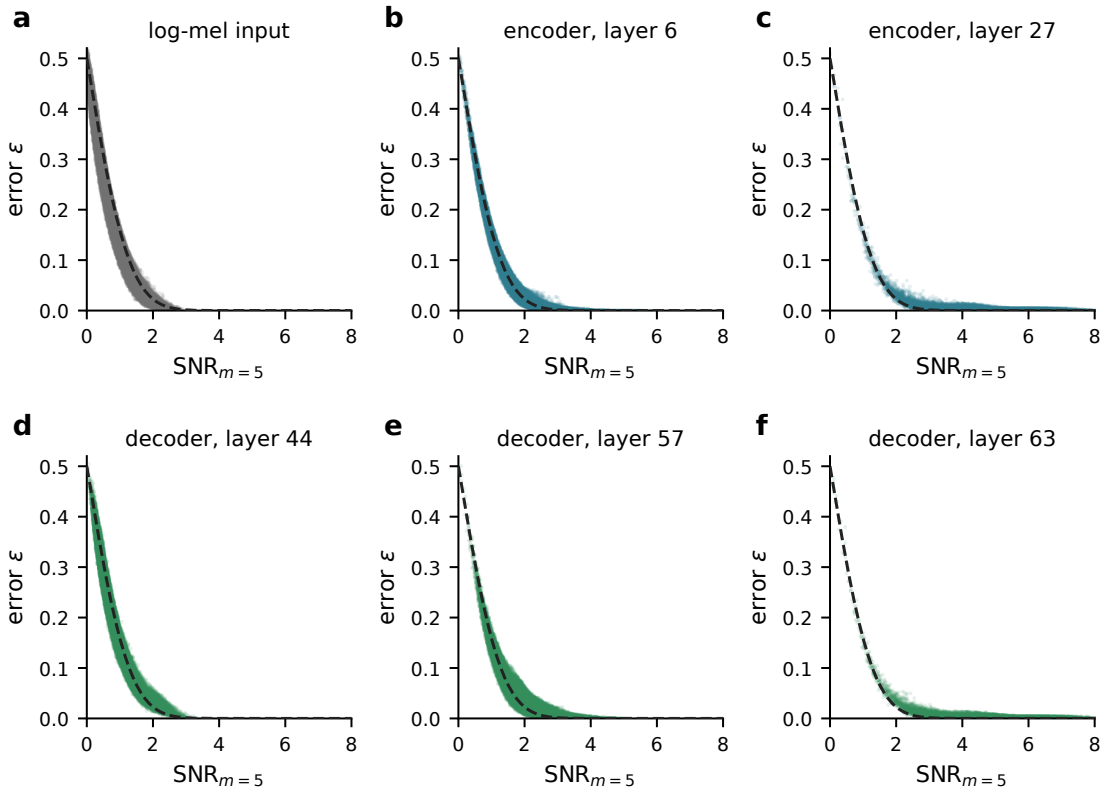


Figure 3.4: Predicted and measured five-shot prototype error in the single-word condition. Each point represents one of the 800×799 ordered word pairs at (a) the log-mel input, (b, c) encoder layers 6 and 27, or (d–f) decoder layers 44, 57, and 63. Measured error ε, estimated over 200 prototype draws, is plotted against $\mathrm{SNR}_{m=5}$ from Eq. B.1. The dashed curve shows the theoretical prediction $\varepsilon = H(\mathrm{SNR})$ from Eq. 2.8. Grey, teal, and green denote the input, encoder, and decoder, respectively.

3.4.3 Manifold geometry predicts few-shot error

For every ordered word pair and processing stage, we compute the signal-to-noise ratio in Eq. 2.7 with $m = 5$ and predict the corresponding error using Eq. 2.8. Figure 3.4 compares measured and predicted errors at representative stages of the input, encoder, and decoder. Across all six stages, the error of each word pair closely follows the theoretical curve. As processing progresses within each stack, the distribution shifts along this curve toward higher signal-to-noise ratios and lower errors.

This agreement allows the signal-to-noise ratio to serve as the primary measure of separability in

the analyses that follow. It retains dynamic range as performance approaches the ceiling and can be decomposed into the geometric terms introduced in Section 3.2.4. Figure 3.5 shows its trajectory across depth alongside the measured error.

The signal-to-noise ratio is 0.73 at the spectrogram and 0.93 after the convolutional stem (Fig. 3.5b). It rises through the encoder to a peak of 5.75 at layer 27, then declines to 4.54 at the encoder output. At the first decoder layer, it returns to 0.78 and remains near 1 through layer 54. It then rises sharply to a second peak of 6.01 at layer 63 before declining to 4.39 at the final layer. Whisper thus exhibits two maxima of word separability: one late in the encoder and another late in the decoder. In both stacks, separability peaks several layers before the final output. Sections 3.5–3.8 identify the geometric terms that produce these trajectories.

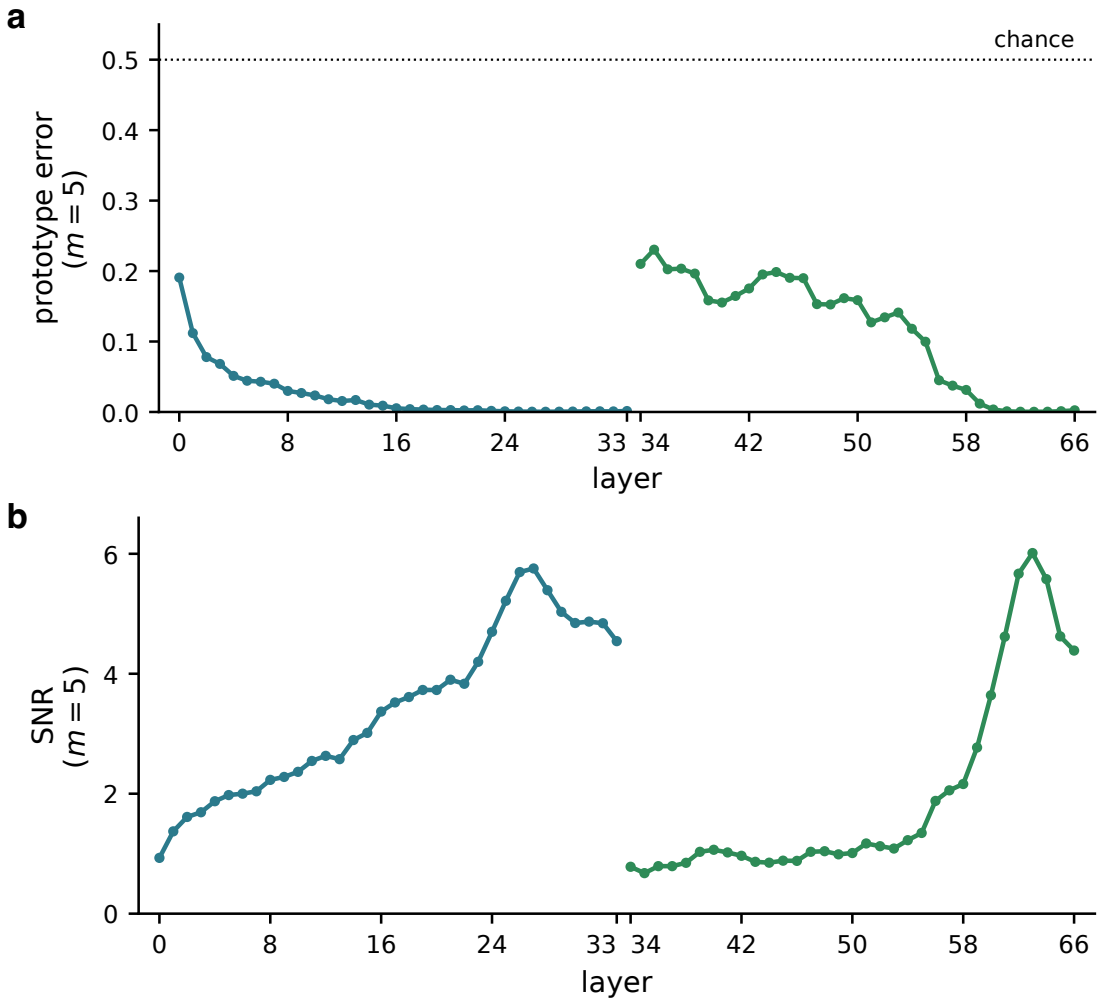


Figure 3.5: Word-manifold separability across Whisper in the single-word condition. (a) Mean five-shot prototype error across all 800×799 ordered word pairs; chance error is 0.5. (b) Mean $\mathrm{SNR}_{m=5}$ from Eq. B.1 across the same pairs. Encoder stages are shown in teal and decoder stages in green. Bands denote $\pm$ one s.e.m. across words and are smaller than the markers in (b).

3.5 Geometric mechanisms of word separability

The signal-to-noise ratio can increase through several geometric transformations defined in Chapter 2. We estimate each term at every layer (Appendix B.3). In the single-word condition, the bias term remains small throughout the model (Appendix B.3.2). We therefore focus on signal, dimensionality, and signal–noise overlap, which drive the main changes in separability across depth (Fig. 3.6).

The initial increase in encoder separability arises from a reorganization of within-word variability rather than an increase in signal. Between the convolutional stem (layer 0) and layer 9, the signal-to-noise ratio increases from 0.93 to 2.28. However, the signal shows no net increase: it falls from 0.98 to 0.61 after the first block and recovers to 0.93 by layer 9 (Fig. 3.6a). Over the same interval, dimensionality triples from 7.6 to 23.3, while signal–noise overlap decreases sevenfold from 0.19 to 0.027, with most of the reduction occurring in the first block (Fig. 3.6b,c). Within-word variability thus becomes higher-dimensional and less aligned with the directions separating word centroids.

Signal amplification dominates the subsequent increase in encoder separability. From layer 9 to the encoder peak at layer 27, the signal-to-noise ratio rises from 2.28 to 5.75, while the signal more than triples from 0.93 to 3.10. The increase is initially gradual but accelerates over the final five layers (Fig. 3.6a). Dimensionality peaks at 29.5 at layer 19 before returning to 22.3 at layer 27, close to its starting value. Its net contribution is therefore small. Signal–noise overlap increases from 0.027 to 0.059, partially opposing the gain in signal. After normalization by signal magnitude, however, overlap stabilizes by layer 13, indicating little subsequent reorientation of the manifold axes. The increase in separability is therefore driven primarily by greater centroid separation relative to manifold radius.

After layer 27, reduced signal and dimensional compression lower encoder separability. By the encoder output at layer 33, the signal-to-noise ratio has declined from 5.75 to 4.54, the signal from

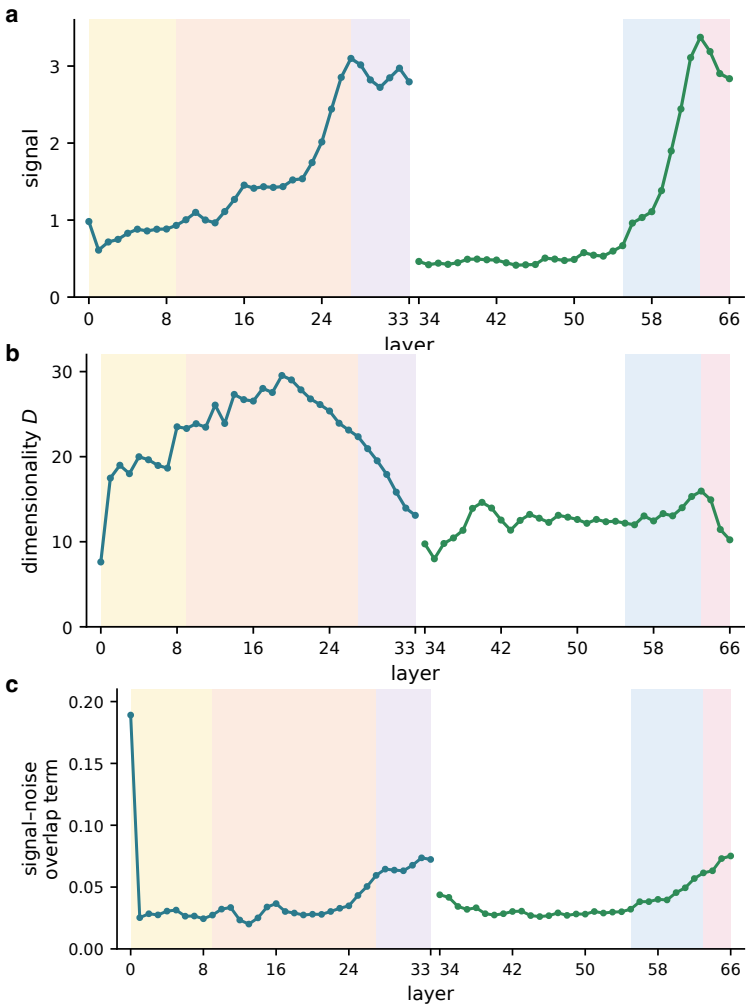


Figure 3.6: Geometric mechanisms of word separability in the single-word condition. (a) Signal, defined as the squared distance between word centroids relative to the manifold radius, $||\Delta \mathbf{x}_0||^2$ in Eq. 2.6. (b) Dimensionality D of the word manifolds (Eq. 2.5). (c) Signal–noise overlap term in Eq. B.1. Encoder and decoder stages are shown in teal and green, respectively. Shaded regions mark the expansion, amplification, and compression regimes described in the text. Lines are means across all 800×799 ordered pairs, or across 800 words for D; bands denote $\pm$ one s.e.m. across words and are smaller than the markers in (a) and (c).

3.10 to 2.79, and dimensionality from 22.3 to 13.1. At the same time, signal–noise overlap increases from 0.059 to 0.072. Within-word variability therefore becomes concentrated in fewer directions, with a greater component along the centroid-separation direction. Together with the modest loss of signal, this compression accounts for the decline from the encoder peak.

Word separability remains low through most of the decoder. Between layers 34 and 55, the signal remains between 0.41 and 0.67, below its spectrogram value of 0.84, while the signal-to-noise ratio ranges from 0.68 to 1.35. A transient improvement occurs between layers 34 and 40, during which the ratio rises from 0.78 to 1.07 without a corresponding increase in signal. As in the early encoder, this improvement reflects an increase in dimensionality, from 9.7 to 14.6, and a decrease in signal–noise overlap, from 0.044 to 0.027. Both changes largely reverse by layer 44.

The late decoder increase is instead driven by signal amplification. From layer 55 to the decoder peak at layer 63, the signal increases fivefold from 0.67 to 3.37, while the signal-to-noise ratio rises from 1.35 to 6.01. Dimensionality also increases, from 12.2 to 16.0, but this favorable change is approximately offset by an increase in signal–noise overlap from 0.032 to 0.061. After the peak, the signal falls to 2.84, dimensionality contracts to 10.2, and overlap rises further to 0.075. These changes reduce the ratio to 4.39 at the decoder output.

Word separability therefore emerges through two distinct geometric transformations. The early encoder expands within-word variability across more dimensions and reduces its alignment with the directions separating words. Later encoder and decoder layers instead amplify centroid separation relative to manifold radius, although the resulting gains are partly opposed by greater signal–noise overlap. After both peaks, weaker signal and dimensional compression reduce separability before the stack output. The principal difference between the two stacks is therefore their use of expansion: the encoder expands and then amplifies, whereas the decoder relies predominantly on amplification. The following sections examine which properties of word pairs predict their separability (Section 3.6), whether phoneme manifolds undergo the same transformations (Section 3.7), and how

preceding linguistic context alters these geometric terms (Section 3.8).

3.6 Acoustic, phonemic, and semantic differences across depth

Having identified the geometric transformations that produce word separability, we next ask what distinctions between words organize that geometry. Specifically, we test whether pairwise separability reflects differences in acoustic duration, phonemic form, or semantic meaning, and how the contributions of these properties change across processing stages. For each word pair, duration difference is the absolute difference between mean durations, phoneme distance is the edit distance between dictionary phoneme sequences, and semantic distance is one minus the cosine similarity between word2vec embeddings (Section 3.10.7). Duration and phoneme distance are examined in the single-word condition, whereas semantic distance is examined both for isolated words and with preceding sentence context.

Because these measures are correlated, we compare pairs with large and small differences in one measure while matching the others. Matching phoneme-count differt acoustic distance additionally additionally; the duration analy additionallysis also controls for mean duration, and the semantic analysis for the mean and difference in word frequency. Within each matched stratum containing at least 100 pairs, the upper and lower 5% of the target measure define the two groups. The resulting contrasts are 255 versus 10 ms for duration, 6.9 versus 2.9 for phoneme distance, and 1.03 versus 0.67 for semantic distance. Standardized differences in the control variables are at most 0.10, except for mean frequency in the semantic analysis (-0.15). Statistical inference uses simultaneous 95% bootstrap intervals across layers (Section 3.10.7).

Duration difference predicts separability primarily near the input to each stack. At layer 0, pairs with large duration differences have a mean signal-to-noise ratio of 1.72, compared with 0.66 for pairs of similar duration (Fig. 3.7a). This advantage diminishes through the early encoder, disap-

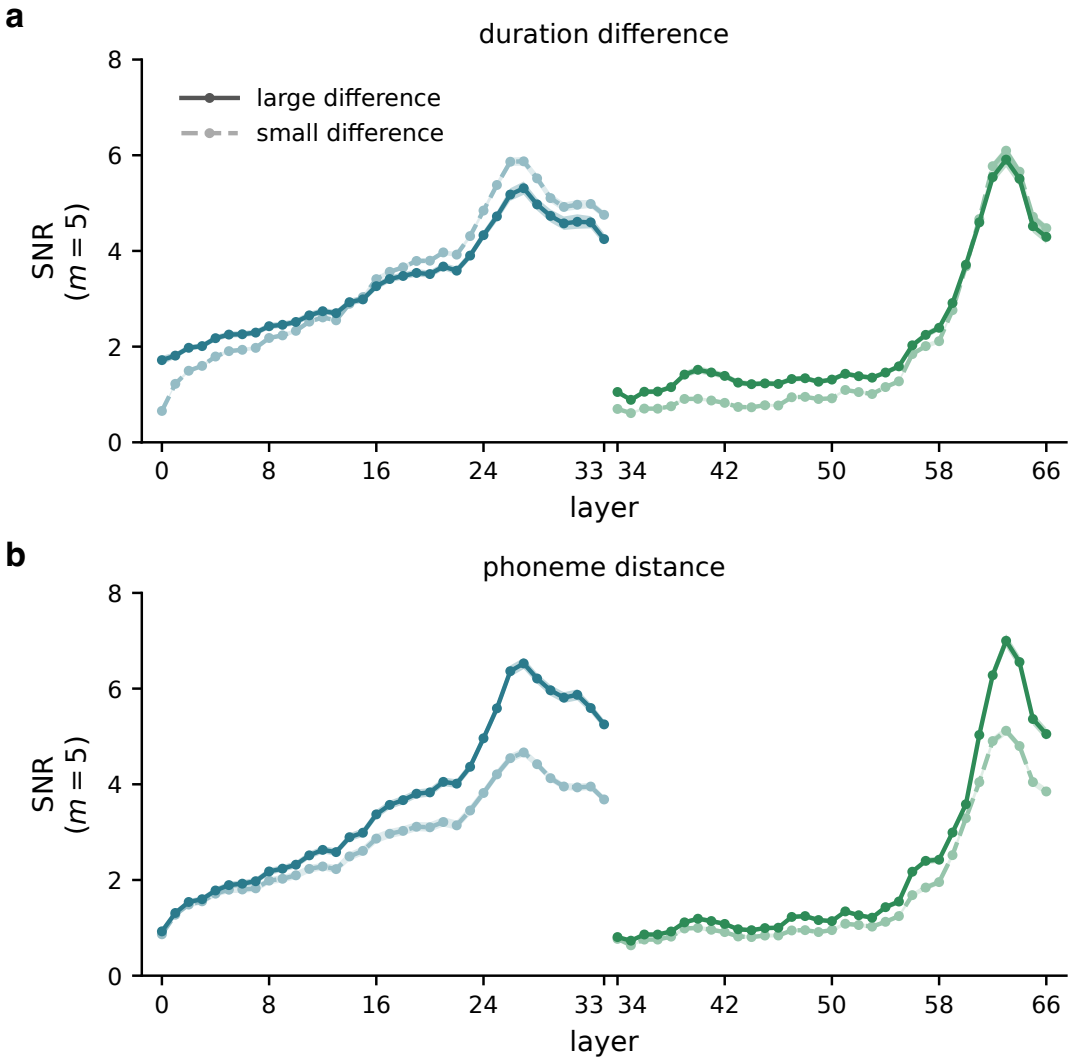


Figure 3.7: Associations of (a) duration difference and (b) phoneme-sequence distance with word separability in the single-word condition. Solid and dashed curves show the mean $\mathrm{SNR}_{m=5}$ for matched word pairs with large and small differences, respectively. The duration analysis contains $15{,}122$ unordered pairs per group (255 versus 10 ms); the phoneme analysis contains $15{,}724$ pairs per group (edit distances of 6.9 versus 2.9). Groups are matched as described in Section 3.10.7. Encoder and decoder stages are shown in teal and green. Bands denote the 2.5th–97.5th percentiles of $10{,}000$ bootstrap samples over word types and are mostly smaller than the markers.

pears after layer 9, and reverses from layer 19 onward. At the encoder peak, the corresponding ratios are 5.31 and 5.87. The duration effect reappears in the early decoder, reaching 1.52 versus 0.91 at layer 40, but disappears before the decoder peak.

The signal remains larger for words that differ in duration through layer 60, indicating that changes in the noise terms offset this separation at later stages (Appendix B.4). Spectrogram distance produces a similar trajectory across the encoder. Duration-related acoustic differences therefore remain present in the representation but cease to determine separability at the depths where word manifolds are most distinct.

Phoneme-sequence distance shows the opposite profile. The matched groups are indistinguishable at layer 0, with ratios of 0.93 and 0.87, but diverge from layer 9 onward (Fig. 3.7b). At the encoder peak, pairs with large phoneme distances reach 6.53, compared with 4.67 for pairs with small distances. The difference remains modest across the decoder plateau and widens during the late increase, reaching 7.00 versus 5.12 at the decoder peak.

This phonemic effect persists at both stack outputs for isolated words. With sentence context, it remains present at both separability peaks and at the decoder output, but not at the encoder output, where the word manifolds collapse (Section 3.8). Because the groups are matched on phoneme count and input acoustic distance, the effect cannot be attributed to word length or gross spectral dissimilarity. It is expressed primarily through the signal term: greater phoneme distance predicts greater centroid separation from encoder layer 15 through decoder layer 56 (Appendix B.4). Phonemic form therefore becomes a principal organizing variable of word geometry at the stages of maximal separability.

Semantic distance predicts separability primarily when preceding sentence context is available. For isolated words, the effect is small: semantically distant pairs exceed similar pairs by only 0.1–0.3 through the late encoder and decoder, reaching 6.03 versus 5.86 at the decoder peak (Fig. 3.8a). This difference is approximately an order of magnitude smaller than the corresponding phonemic

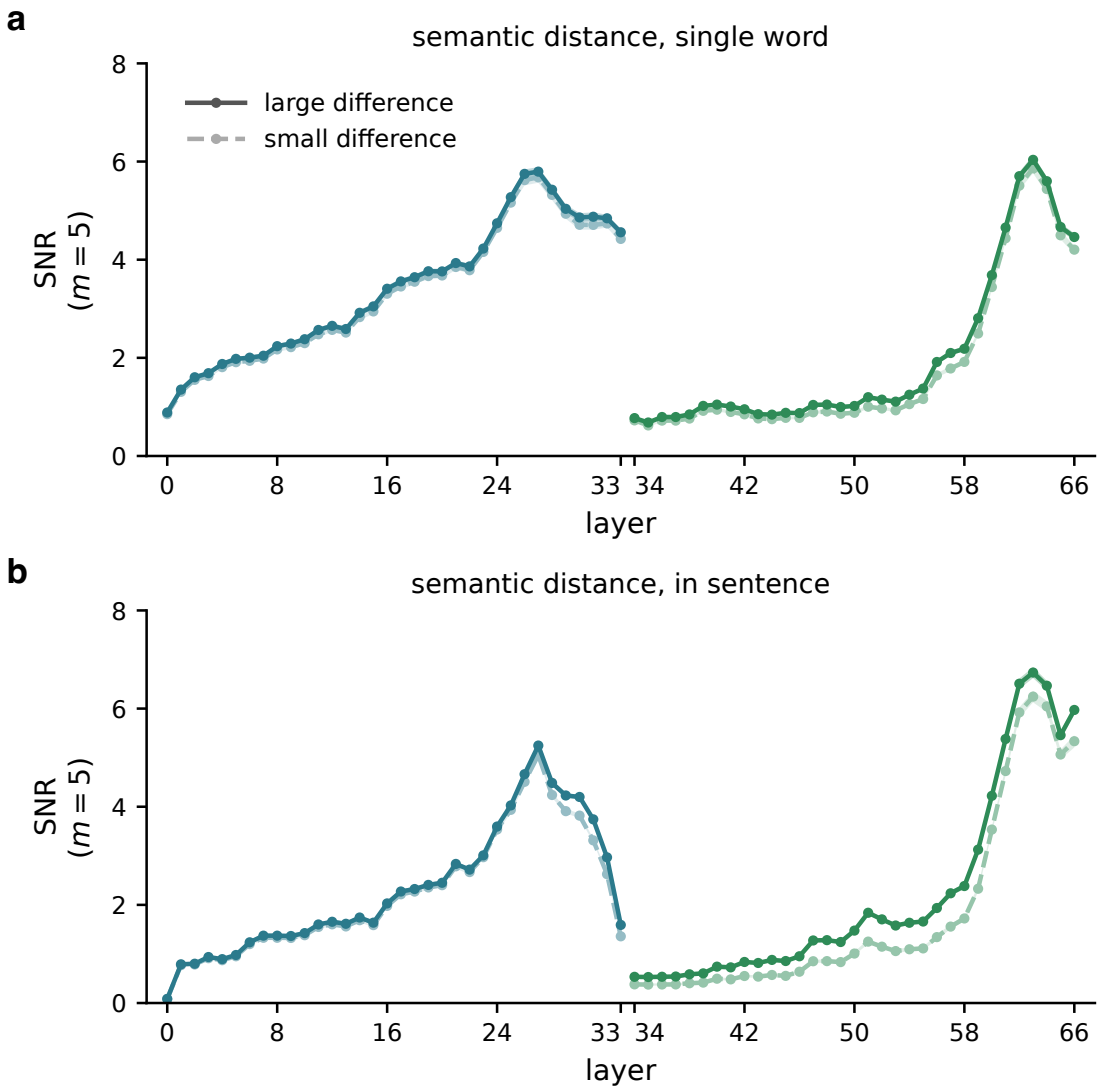


Figure 3.8: Association between semantic distance and word separability in (a) the single-word and (b) the in-sentence condition. Semantic distance is one minus the cosine similarity between word2vec embeddings. Solid and dashed curves show the mean $\mathrm{SNR}_{m=5}$ for matched word pairs with large and small semantic distances, respectively ($13{,}954$ unordered pairs per group; mean distances 1.03 and 0.67). Groups are matched on the acoustic and phonemic measures in Fig. 3.7 and on word frequency (Section 3.10.7). Encoder and decoder stages are shown in teal and green. Bands denote the 2.5th–97.5th percentiles of $10{,}000$ bootstrap samples over word types and are mostly smaller than the markers.

effect.

With sentence context, the semantic effect is two to three times larger and significant from layer 26 onward (Fig. 3.8b). It grows across the decoder plateau, reaches its maximum shortly before the decoder separability peak (3.13 versus 2.33 at layer 59), and persists at the decoder output (5.97 versus 5.34). Alternative measures based on BERT distance and grammatical class yield smaller effects concentrated on the decoder plateau (Appendix B.4).

The properties associated with word separability thus change across depth, but not through a simple replacement of one code by another. Duration matters near the input to each stack but not at the separability peaks. Phonemic distance emerges through the encoder, is strongest at both peaks, and remains evident at the stack outputs. Semantic distance has little effect for isolated words but becomes more prominent in the decoder when preceding context is available. At Whisper's most separable stages, word-manifold geometry is therefore organized primarily by phonemic form, with context introducing a weaker semantic organization.

3.7 Phoneme manifolds follow a distinct geometric trajectory

The preceding analyses characterize how Whisper constructs separable word representations, but words are composed of smaller linguistic units. Phonemes are the contrastive sound categories that distinguish one word from another, and speech recognition must recover these distinctions from continuous acoustic input. The transformation from acoustics to words may therefore pass through an intermediate representation organized around phoneme identity. Examining phoneme manifolds allows us to determine whether such a representation emerges within the encoder and how its trajectory relates to the later construction of word separability.

The acoustic realization of a phoneme varies with the speaker, surrounding sounds, word, and sentence in which it occurs. A phoneme manifold collects these diverse realizations into a single

category, making it possible to measure whether the encoder separates phoneme identity while accommodating variation in its acoustic and linguistic context. Its geometry therefore reveals not only when phoneme identity becomes separable, but also how the encoder organizes the variability within each phoneme.

We consider 39 phoneme manifolds that contain 200 occurrences sampled from distinct TIMIT sentences and across different words, speakers, and phonetic contexts (Section 3.10.5; Fig. 3.9a). Because a phoneme spans only a few encoder positions and has no direct counterpart in the decoder token sequence, the analysis is restricted to the encoder. The geometric quantities, five-shot prototype readout, and linear probe are identical to those used for words. Comparisons between phonemes and words concern their trajectories across depth.

Phoneme separability follows a nonmonotonic trajectory that peaks in the middle of the encoder. Prototype error is 0.14 on the spectrogram, rises to 0.41 after the convolutional stem, and returns to 0.15 after the first encoder block (Fig. 3.9c). It then falls to a minimum of 0.05 at layer 21 before rising sharply from layer 27 onward, reaching 0.23 at the encoder output—higher than at the spectrogram.

At every layer, pairwise error follows the theoretical relation $\varepsilon = H(\mathrm{SNR})$ (Fig. 3.9b; Appendix B.5.3). The mean signal-to-noise ratio rises from 1.28 at layer 1 to a maximum of 1.89 at layer 18, remains between 1.68 and 1.89 through layer 26, and then falls to 0.75 at the encoder output, below its spectrogram value of 1.31 (Fig. 3.9d).

The linear probe does not reveal this late decline. Its 39-way accuracy rises from 0.49 after the convolutional stem to 0.88 at layer 21 and remains between 0.88 and 0.90 thereafter, while its classification margin continues to increase (Appendix B.5.4).

The increase in phoneme separability from layer 1 to layer 18 is produced by an increase in signal (Figure 3.10). The signal rises from 0.37 to 0.70 while the ratio rises from 1.28 to 1.89 (Figure 3.10a). Dimensionality does not contribute: D is 32 at layer 1, its maximum before the late rise,

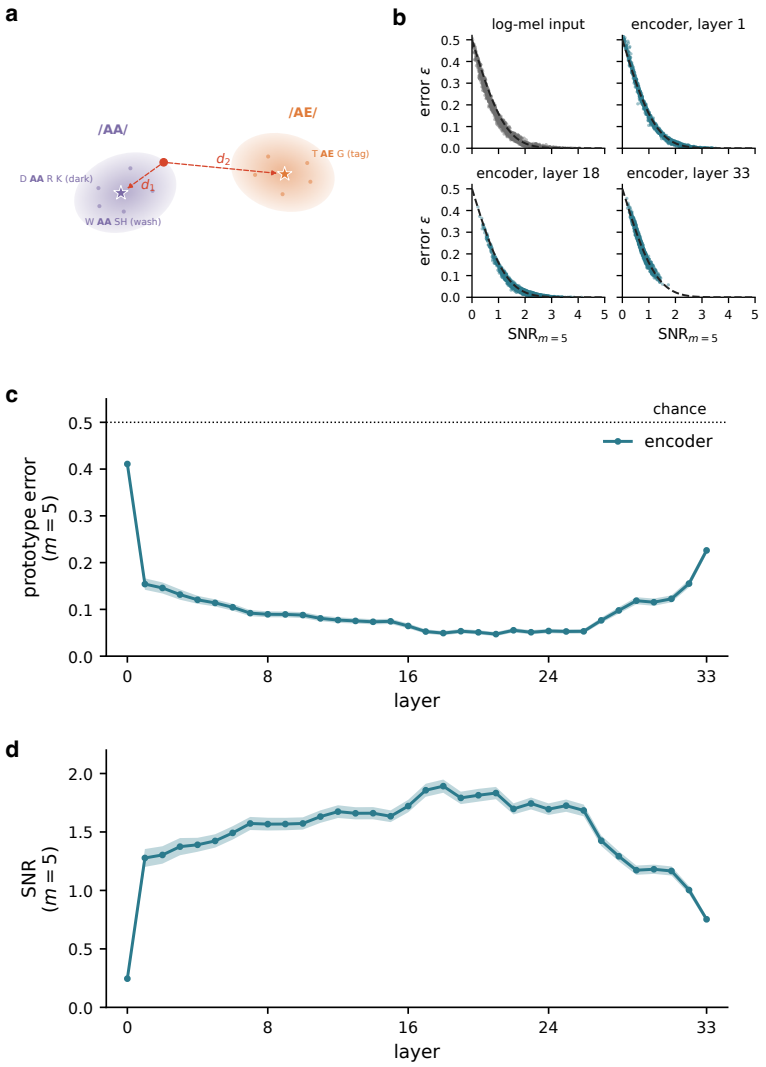


Figure 3.9: Phoneme separability across the Whisper encoder. (a) The five-shot prototype readout on two phoneme manifolds. Each point is one occurrence of the phoneme from a different TIMIT sentence, word, and speaker; three are labeled with the phone sequence of their word, the target phoneme in bold. Five references per phoneme (faded) are averaged into its prototype (star), and a held-out occurrence of /AA/ (red) is assigned to the nearer prototype, at distances d_1 and d_2. (b) Measured error of the readout against the signal-to-noise ratio $\mathrm{SNR}_{m=5}$ of Eq. B.1 for all 39×38 ordered phoneme pairs at the log-mel input (grey) and at encoder layers $1, 18$ and 33 (teal); the black dashed curve is the prediction $\varepsilon = H(\mathrm{SNR})$ of Eq. 2.8. Mean measured and predicted error: 0.138 and 0.153 at the input, 0.154 and 0.155 at layer 1, 0.049 and 0.053 at layer 18, 0.226 and 0.231 at layer 33. (c) Mean error over all ordered pairs against encoder layer; chance is 0.5. (d) Mean $\mathrm{SNR}_{m=5}$ over the same pairs. In (c) and (d) the line is the mean and the band $\pm$ one s.e.m. across the 39 phoneme classes; no decoder representation is defined for a phoneme (Section 3.10.5).

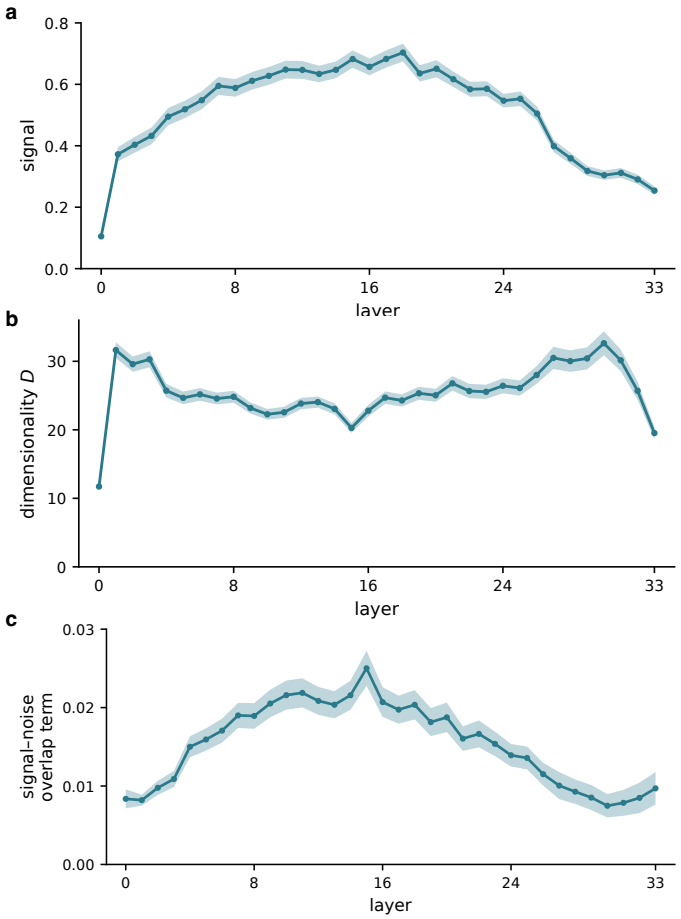


Figure 3.10: Geometry of phoneme manifolds across the Whisper encoder. (a) Signal, the squared separation between two phoneme centroids relative to the manifold radius, $\|\Delta \mathbf{x}_0\|^2$ of Eq. 2.6, mean over all 39×38 ordered phoneme pairs against the encoder layer. (b) Dimensionality D of the phoneme manifolds, Eq. 2.5, mean over the 39 phonemes. (c) The signal–noise overlap term of Eq. B.1 at $m = 5$, mean over the same pairs as (a). Lines are the mean and bands $\pm$ one s.e.m. across the 39 phoneme classes.

and decreases to 24 at layer 18 (Figure 3.10b), while the signal–noise overlap term rises with the signal, from 0.008 to 0.020 (Figure 3.10c). The gain is therefore amplification of the centroid separation, partly offset by the overlap it carries with it. From layer 18 to layer 26, the ratio remains between 1.68 and 1.89 while the signal decreases to 0.51, the overlap term falls to 0.012 and D rises to 28.

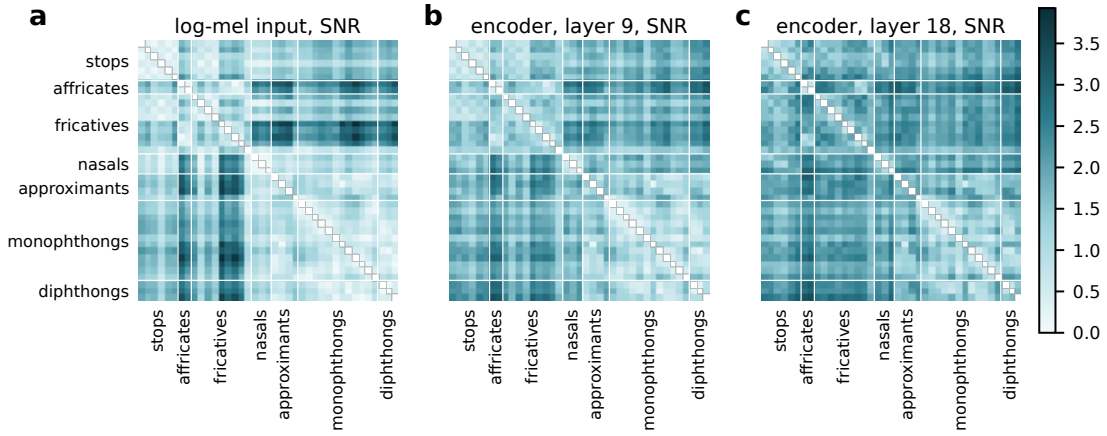


Figure 3.11: Pairwise phoneme separability at (a) the log-mel input, (b) encoder layer 9, and (c) encoder layer 18. Each matrix shows $\mathrm{SNR}_{m=5}$ for all pairs of the 39 phonemes, averaged across the two directions of each pair. Phonemes are grouped by manner and ordered within each group by place and voicing; vowels follow the vowel quadrilateral (Section 3.10.5). All panels share a common color scale, with darker cells indicating greater separability. Mean ratios are 1.31, 1.57, and 1.89, respectively. Within-manner pairs have 0.47, 0.56, and 0.69 times the separability of between-manner pairs at the three stages. Fully labeled matrices are provided in Figures B.8–B.10.

After the broad maximum through layer 26, phoneme separability declines sharply. The signal falls from its peak of 0.70 to 0.25 at the encoder output, and the ratio falls with it to 0.75 (Figure 3.10a). The decline is not initially caused by compression: dimensionality rises to 33 at layer 30 even as the ratio decreases, before falling to 20 after the final layer normalization (Figure 3.10b). The encoder output therefore has a lower phoneme signal-to-noise ratio than the spectrogram, 0.75 against 1.31, despite representing phoneme variability in more dimensions. The late loss of separability is driven primarily by a reduction in centroid separation relative to the manifold radius.

The mean signal-to-noise ratio does not reveal which phoneme distinctions the encoder resolves. Phonemes are defined by articulatory features: consonants by manner, place, and voicing, and vowels by tongue height, backness, rounding, and tenseness (Section 3.10.5). These features also organize human perceptual confusions in noise [103] and cortical responses in the superior temporal gyrus [8]. We therefore examine how pairwise separability is organized by articulatory structure.

In the spectrogram, pairwise separability is dominated by broad acoustic differences (Fig. 3.11a).

Fricatives and affricates, distinguished by their frication noise, are readily separated from other phoneme classes. By contrast, phonemes within the same manner class—including pairs of stops, nasals, or vowels—are poorly separated. On average, within-class pairs have only 0.47 times the separability of between-class pairs.

The middle encoder preferentially increases the separability of these initially confusable contrasts. Between layers 1 and 18, the mean ratio rises from 0.48 to 0.84 for phonemes differing only in voicing, from 0.68 to 1.57 for those differing only in place, and from 0.43 to 0.86 for vowel pairs differing in one feature. By comparison, the already separable consonant–vowel pairs increase by a factor of only 1.32. Consequently, within-class separability reaches 0.69 of the between-class value at layer 18. The least separable pairs remain neighboring vowels and consonants that differ only in voicing, but the overall pairwise geometry is substantially more uniform than at the input.

During the subsequent decline, within-class pairs lose slightly less separability than between-class pairs, raising their ratio to 0.80 at the encoder output. The middle encoder therefore resolves many of the fine phonemic contrasts that are poorly separated in the spectrogram, even though this context-independent phoneme geometry is not preserved at the output.

The phoneme and word trajectories can now be compared along their shared encoder depth. During layers 0–9, word manifolds gain separability through dimensional expansion, whereas phoneme manifolds are already high-dimensional and improve through increasing signal. Phoneme separability peaks at layer 18, where the word signal is 1.43, less than half its eventual peak of 3.10. The most separable phoneme geometry therefore emerges before the principal amplification of word separation.

The rapid increase in word signal between layers 22 and 27 coincides with increasing organization of phoneme responses by lexical context (Appendix B.5.6). The distance between two occurrences of the same phoneme within the same word, relative to occurrences in different words, decreases from 0.99 at layer 1 to 0.59 at layer 27. When occurrences also share the sentence, this

ratio reaches 0.16, whereas it remains near 1 after permuting word labels. Because occurrences of a phoneme within the same word also share neighboring phonemes, this measure combines lexical identity with local phonetic context (Section 3.9.5). Nevertheless, the result indicates that within-phoneme variability becomes increasingly structured by the word and its context as word separation is amplified.

Phoneme separability begins to decline at layer 27, where word separability reaches its maximum. This temporal correspondence does not establish a trade-off. Still, it it it is consistent with a transition from a relatively context-independent phoneme geometry to representations organized more strongly by words and their contexts.

The geometric route to separability therefore depends on the linguistic unit, not only on encoder depth. Word separability initially increases through dimensional expansion and later through signal amplification. Phoneme manifolds, by contrast, become high-dimensional by the first encoder block, but this expansion produces no net gain in separability; subsequent improvement is driven by signal amplification. The resulting geometry resolves fine phonemic contrasts in the middle encoder, then deteriorates as word-conditioned structure emerges. Whisper therefore constructs context-independent phoneme separability as a transient intermediate representation rather than preserving it at the encoder output.

3.8 Context modulates word geometry in the decoder

The preceding analyses establish how Whisper separates words presented without preceding speech. In continuous speech, however, a word is embedded in a sentence whose earlier sounds and words provide acoustic and linguistic context. This context could alter separability by changing the distance between word centroids, adding within-word variability, or changing how that variability is distributed relative to word-separating directions. We therefore use the geometric decomposition

to determine where preceding context enters the representation and whether it impairs or improves word separability in the encoder and decoder.

In the in-sentence condition, Whisper receives the sentence audio from its onset through the target word, and the decoder input contains the task prefix followed by the ground-truth tokens preceding that word (Section 3.3). The position-matched control presents the identical target waveform at the same location in the input window but replaces the preceding speech with silence; its decoder receives only the task prefix. The encoder comparison therefore isolates the effect of preceding speech, whereas the decoder comparison captures the combined effects of that speech and its transcript. All 160,000 word occurrences are included in both conditions, irrespective of whether Whisper transcribes them correctly (Appendix B.6.1).

The isolated-word condition used in the preceding sections places the target at the beginning of the input window and therefore differs in absolute position as well as context; its comparison with the two position-matched conditions is reported in Appendix B.6.2. Figure 3.12 shows prototype error and signal-to-noise ratio for the in-sentence and control conditions, and Figure 3.13 shows the corresponding signal, dimensionality, and signal–noise overlap. The prediction $\varepsilon = H(\mathrm{SNR})$ of Section 3.4.3 holds for in-sentence pairs (Figure 3.12c).

Through most of the encoder, preceding speech has little effect on word-manifold geometry. The in-sentence signal-to-noise ratio closely follows the position-matched control through layer 30: the difference is at most 0.6 throughout and only 0.19 at their shared peak at layer 27 (5.19 versus 5.00; Fig. 3.12b). The geometric components are similarly close. At layer 27, the signal is 2.18 versus 2.42, and dimensionality is 20.1 versus 20.7 (Fig. 3.13a,b); manifold radius and centroid separation also differ little (Appendix B.6.3). Thus, although every encoder block can integrate information across the preceding audio, that context has limited influence on the geometry of target-word identity until the final encoder layers.

This pattern changes abruptly over the final three encoder layers. From the shared peak at layer

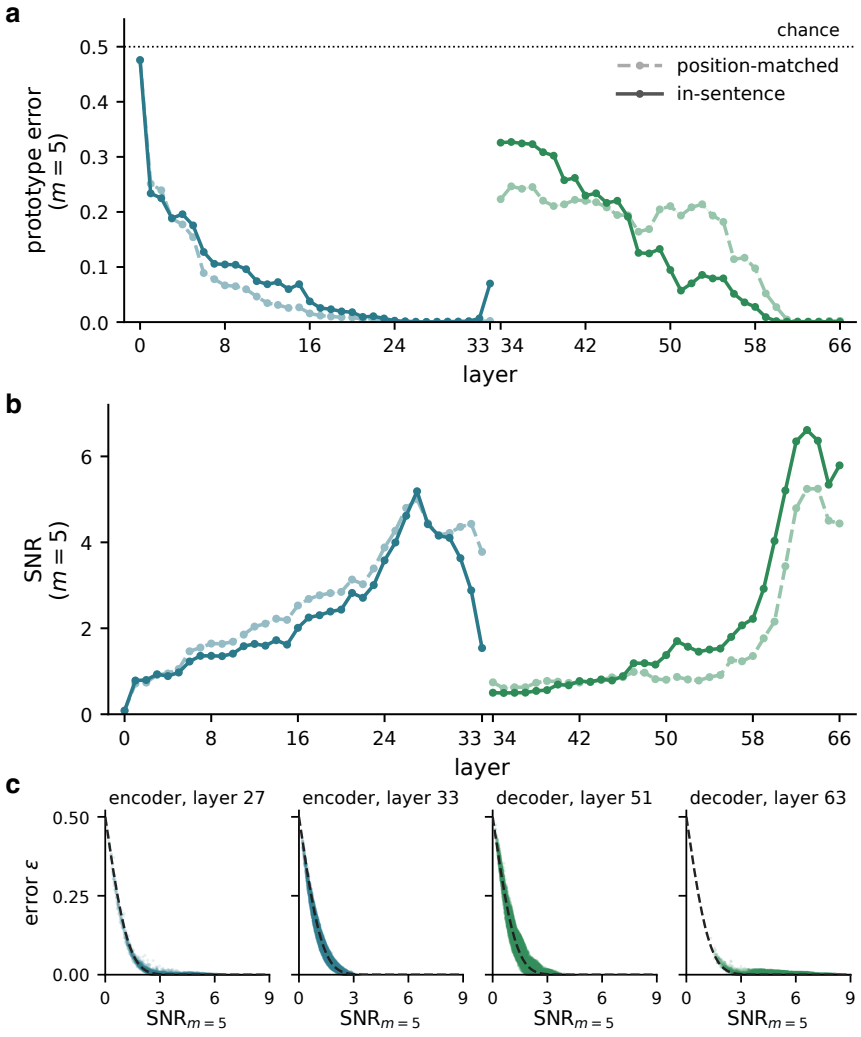


Figure 3.12: Effects of preceding sentence context on word separability. Solid curves show the in-sentence condition, in which the target word is preceded by sentence audio and, in the decoder, by its ground-truth transcript. Dashed curves show the position-matched control, in which the identical target waveform occurs at the same position but is preceded by silence, and the decoder receives only the task prefix. Thus, the encoder comparison isolates the effect of preceding audio, whereas the decoder comparison includes both preceding audio and transcript tokens. (a) Mean five-shot prototype error across all 800×799 ordered word pairs; chance error is 0.5. (b) Mean $\mathrm{SNR}_{m=5}$ across the same pairs. Context has little effect through encoder layer 30, reduces the ratio from 3.78 to 1.54 at the encoder output, and increases it from 5.24 to 6.61 at the decoder peak. (c) Measured error against $\mathrm{SNR}_{m=5}$ for in-sentence word pairs at encoder layers 27 and 33 and decoder layers 51 and 63. The dashed black curve shows the theoretical prediction $\varepsilon = H(\mathrm{SNR})$. Encoder and decoder stages are shown in teal and green; bands in (a,b) denote $\pm$ one s.e.m. across words.

27, the in-sentence signal-to-noise ratio falls to 1.54 at the encoder output, compared with 3.78 in the control, and prototype error rises to 0.070 versus 2.2×10^{-3} (Fig. 3.12a,b). The decline reflects a large increase in within-word variability: the in-sentence manifold radius grows seventeen-fold, nearly twice the increase in the control, while centroid separation changes similarly in the two conditions (Appendix B.6.3). Consequently, the signal falls to less than half the control value, and dimensionality decreases to 7.1 versus 12.7 (Fig. 3.13a,b). Preceding speech therefore introduces a large component of variability concentrated along relatively few directions, sharply reducing word separability at the encoder output.

Context initially reduces separability in the decoder. At layer 34, each in-sentence representation is conditioned on a different preceding utterance and transcript, whereas the control decoder receives the same task prefix for every occurrence. The in-sentence manifolds consequently have eleven times the radius of the control manifolds and approximately half the signal, yielding a lower signal-to-noise ratio (0.50 versus 0.74) and higher error (0.33 versus 0.22; Figs. 3.12a,b and 3.13a). Unlike the low-dimensional variability introduced at the encoder output, however, decoder contextual variability is high-dimensional from the first layer: $D = 21.9$ in the in-sentence condition, compared with 11.4 in the control.

The effect of this variability reverses across the decoder plateau. In the in-sentence condition, dimensionality rises to 44.9 by layer 52, while remaining near 13 in the control (Fig. 3.13b). Signal–noise overlap also becomes substantially lower with context (Fig. 3.13c). The in-sentence signal-to-noise ratio is therefore lower than the control through layer 41, comparable through layer 46, and higher from layer 47 onward. By layer 51, it reaches 1.70, compared with 0.87 in the control, even though the in-sentence signal remains lower. The advantage arises because contextual variability is distributed across more dimensions and is less aligned with the directions separating words.

The same geometry produces a larger contextual advantage in the late decoder. At layer 63, the in-sentence signal-to-noise ratio reaches 6.61, compared with 5.24 in the control, despite a smaller

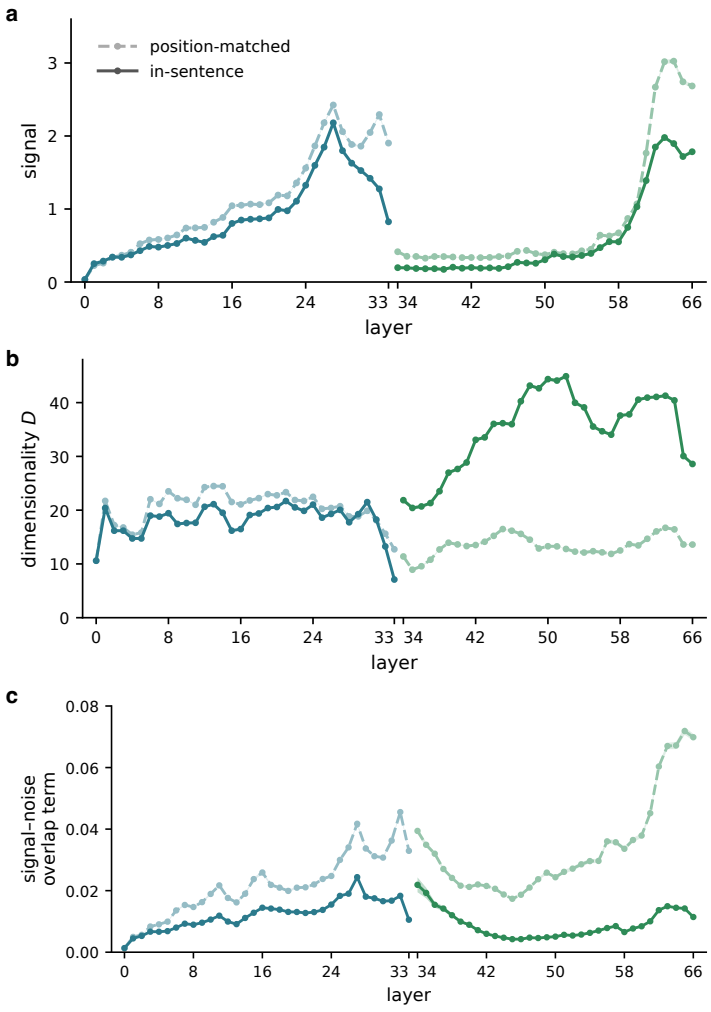


Figure 3.13: Geometric effects of preceding sentence context on word separability. Solid curves show the in-sentence condition and dashed curves the position-matched control. (a) Signal, defined as squared centroid separation relative to manifold radius. (b) Dimensionality D. (c) Signal–noise overlap. Context has little effect on these quantities through most of the encoder but reduces signal and dimensionality over its final layers. In the decoder, the in-sentence condition has lower signal but higher dimensionality and lower signal–noise overlap. Encoder and decoder stages are shown in teal and green. Lines are means across ordered word pairs, or across words for D; bands denote $\pm$ one s.e.m. across words.

signal (1.98 versus 3.02). Dimensionality is substantially higher with context (41.3 versus 16.7), while signal–noise overlap is approximately one quarter of the control value (Fig. 3.13). The advantage persists at the decoder output, where the ratios are 5.79 and 4.44, respectively, and the in-sentence prototype error is fourfold lower (Fig. 3.12a; Appendix B.6.1). Thus, context increases late decoder separability not by strengthening the signal, but by distributing variability across many directions that overlap weakly with word-separating directions.

Preceding context therefore has distinct geometric effects in the encoder and decoder. It has little influence through most of the encoder but introduces large, low-dimensional variability at the encoder output, reducing word separability. In the decoder, context initially enlarges the manifolds and impairs separability, but this variability becomes increasingly high-dimensional and weakly aligned with word-separating directions. The decoder thereby achieves greater late-stage separability with context despite maintaining a weaker signal.

The contextual advantage emerges at decoder layer 47, before the late amplification of the word signal, and coincides with the emergence of semantic organization reported in Figure 3.8b. This temporal correspondence is consistent with predictive information from the preceding transcript contributing to decoder separability. Likewise, the abrupt change at the encoder output may reflect preparation of its representations for decoder cross-attention. These interpretations remain hypotheses because the present comparison does not isolate the contributions of preceding audio and transcript tokens within the decoder; they are revisited in Section 3.9.3.

3.9 Discussion

3.9.1 What geometry reveals beyond probe accuracy

A representation can appear stable under one readout while its separability continues to change. Across decoder layers 57–66, linear-probe accuracy remains between 0.94 and 0.96, suggesting

that word information has reached a plateau (Section 3.4.1). Over the same layers, however, five-shot prototype error reaches its minimum at layer 63 and then increases approximately eightfold by the decoder output (Section 3.4.2). The signal-to-noise ratio of Eq. 2.7 predicts this pairwise error across processing stages, while its constituent terms identify the geometric changes underlying the trajectory. The framework therefore resolves changes that are obscured once linear accuracy approaches ceiling.

Three results illustrate why this decomposition is informative. First, word separability improves over encoder layers 0–9 without a net increase in signal. The signal-to-noise ratio rises from 0.93 to 2.28 because within-word variability spreads across approximately three times as many dimensions and becomes less aligned with the directions separating word centroids (Section 3.5). A reduction in error therefore need not reflect greater centroid separation. Second, phoneme dimensionality increases from 4.3 at the spectrogram to 32 after the first encoder block, yet the signal-to-noise ratio remains nearly unchanged because the signal decreases concurrently (Section 3.7). A stable error can thus conceal a substantial reorganization of variability.

Third, a representation with weaker signal can nevertheless be more separable. At decoder layer 63, words presented in sentence context have lower signal than the same words preceded by silence (1.98 versus 3.02) but a higher signal-to-noise ratio (6.61 versus 5.24; Section 3.8). Context increases within-word variability, but distributes it across more dimensions and away from the directions separating words. Centroid separation alone would therefore rank the two conditions incorrectly. Across these examples, prototype error quantifies whether two manifolds are separable, whereas the geometric terms specify why.

These geometric contributions should be interpreted as properties of the representation under the chosen readout, not as independently established causal operations. Whisper is trained to predict transcript tokens, not to optimize few-shot prototype classification. Accordingly, word separability peaks within both the encoder and decoder and declines before their outputs, while phoneme

separability peaks in the middle encoder and falls below its input value at the encoder output. These declines do not imply that later layers discard information needed for transcription; they show that the geometry favored by the prototype readout is not preserved monotonically as the model approaches its learned output.

Chapter 2 used the same framework to compare geometric solutions across architectures. The present chapter instead uses it to resolve successive transformations within a single model. In both cases, the principal contribution is a level of description between scalar performance and individual units: category separability is expressed in terms of the magnitude, dimensionality, and orientation of population variability. The following sections use this description to interpret the progression from phoneme to word geometry and the reorganization produced by sentence context.

3.9.2 From phoneme to word geometry

The encoder does not apply a common geometric transformation to every linguistic unit. Word and phoneme manifolds follow different trajectories despite being measured along the same sequence of layers. Word separability initially improves through the expansion of within-word variability and later through amplification of centroid separation. Phoneme manifolds, by contrast, become high-dimensional by the first encoder block, but this expansion is accompanied by a loss of signal and produces no net improvement in separability. Their subsequent improvement is driven primarily by signal amplification. The geometric route to separability therefore depends on the unit represented, not only on processing depth.

The middle encoder develops a context-tolerant representation of phoneme identity. Phoneme separability reaches a broad maximum over layers 17–26, where the encoder preferentially resolves contrasts that are poorly separated in the spectrogram. These include phonemes within the same manner class, such as stops that differ only in voicing or place, and neighboring vowels that differ in a single articulatory feature. Broad acoustic classes are already distinguishable at the input; the

principal contribution of the middle encoder is to increase the separability of finer phonemic distinctions. The resulting geometry remains structured by articulatory similarity—the least separable pairs are still neighboring vowels and voicing contrasts—but is substantially more uniform than the acoustic geometry from which it develops.

Phoneme separability reaches its maximum before the principal amplification of word separation. At phoneme layer 18, the word signal is 1.43, less than half its subsequent peak of 3.10 at layer 27. As the word signal rises steeply over the intervening layers, phoneme separability begins to decline and within-phoneme variability becomes increasingly organized by the word and its local phonetic context (Section 3.7). This temporal correspondence is consistent with a progression from representations organized primarily by phoneme identity toward representations organized more strongly by larger linguistic units. It does not, however, establish that word formation causally displaces phoneme information. The word and phoneme analyses use different corpora and pooling procedures, and occurrences of a phoneme within the same word also share neighboring sounds. The results therefore establish an ordering across encoder depth, not a direct conversion of one representational code into another.

The properties associated with word separability provide a complementary view of this progression. Duration difference is most strongly associated with separability near the input to each stack, whereas phoneme-sequence distance emerges through the encoder and is largest at the encoder and decoder separability peaks. Semantic distance has little association with the geometry of isolated words but becomes more prominent in the decoder when the preceding sentence is available (Section 3.6). These effects do not form a strict succession in which acoustic, phonemic, and semantic codes replace one another. Rather, different properties contribute with different strengths across depth: acoustic structure remains represented after its effect on separability diminishes, phonemic form dominates at the stages of maximal word separation, and sentence context adds a weaker semantic organization in the decoder.

This progression is broadly consistent with layerwise analyses of self-supervised speech models, which have reported a transition from acoustic to phonetic and lexical information followed, in some models, by a decline toward the final layers[88,89]. Those trajectories vary across architectures and training objectives, particularly after fine-tuning for speech recognition. The present analysis therefore does not establish a universal hierarchy. It adds a geometric account of how such a hierarchy can arise: intermediate phoneme separability is produced by amplification of fine phonemic contrasts, whereas word separability emerges through an earlier redistribution of variability followed by later amplification of word-level signal. In Whisper, separable phoneme identity is thus a prominent intermediate geometry rather than the dominant organization of the encoder output.

3.9.3 Sentence context reorganizes word variability

Sentence context affects separability through the geometry of the variability it introduces. In the position-matched comparison, the target waveform and its location are identical; only the preceding audio and, in the decoder, its transcript differ. Context has little effect on word-manifold geometry through most of the encoder, but sharply reduces separability at the encoder output. In the late decoder, it instead increases separability despite producing a weaker signal.

At the encoder output, preceding speech enlarges the word manifolds without a corresponding increase in centroid separation. This variability is concentrated along relatively few directions, reducing the signal-to-noise ratio from 3.78 after silence to 1.54 in sentence context. The effect shows that context alters the geometry of target-word identity in the final encoder layers. It does not show that earlier layers contain no contextual information, only that such information has little effect on the geometric quantities measured here.

In the decoder, contextual variability is larger but distributed more favorably. It initially reduces separability, but across later layers it spreads over more dimensions and becomes less aligned with word-separating directions. At the decoder peak, the contextualized manifolds have a higher signal-

to-noise ratio than the controls (6.61 versus 5.24), despite having lower signal. The decoder therefore improves word separability by organizing contextual variability away from the directions that distinguish word identity.

This result extends previous observations that representations of the same word become increasingly context dependent across the layers of text models[91]. Increased contextual variability does not necessarily make categories less separable: its effect depends on how many dimensions it occupies and how those dimensions align with category separation. This differs from syntactic-category manifolds in text encoders, which become separable through reductions in radius and dimensionality[92]. In Whisper, increased dimensionality instead contributes to contextual word separability.

The decoder comparison combines two sources of context. In the in-sentence condition, the model receives both the preceding audio and its ground-truth transcript, whereas the control contains preceding silence and only the task prefix. The decoder effect therefore cannot be attributed separately to acoustic context, linguistic prediction, or their interaction. The accompanying increase in transcription accuracy is consistent with the established benefit of sentence context[84], but it should be interpreted as the benefit available when the preceding transcript is correct.

3.9.4 Model-derived predictions for neural populations

The geometric quantities studied here can also be estimated from neural population responses. Manifold geometry has previously been used to characterize object representations in cortex[14], and phonetic features are reflected in population responses in superior temporal gyrus[8]. Moreover, neural-predictivity studies suggest a coarse correspondence between speech-model depth and stages of cortical processing (Section 3.2.2). These findings do not imply a one-to-one mapping between Whisper layers and brain regions, but they make the model's geometric trajectories experimentally testable.

If neural populations follow the sequence observed in Whisper, early improvements in word

separability should occur through expansion: within-word dimensionality should increase and signal–noise overlap should decrease with little change in signal. Later improvements should instead be dominated by increasing centroid separation relative to manifold radius. Phoneme separability should peak before word separability, and the phoneme pairs that remain least separable should differ in few articulatory features. In later language-responsive populations, word separation should increasingly reflect phoneme-sequence distance. Under sentence context, contextual variability should occupy more dimensions and align less strongly with word-separating directions than in earlier sensory populations.

These are model-derived hypotheses rather than claims about cortical organization. They would be falsified by, for example, an early neural population in which word separability improves primarily through signal amplification, a hierarchy in which word separability peaks before phoneme separability, or a later population in which context increases variability only along word-separating directions. The predictions concern the ordering of geometric transformations, not their numerical values or a fixed correspondence between individual layers and cortical areas.

Testing them would require neural responses to many occurrences of each phoneme or word across speakers and acoustic contexts. Intracranial recordings from auditory, temporal, and frontal language regions could define a manifold from the population response to each occurrence, with populations ordered anatomically or by response latency. Presenting identical target waveforms in isolation and after sentence context would isolate contextual modulation. The prototype readout and estimator of Appendix B.3 could then be applied directly. Such an experiment would require sufficiently large simultaneously recorded populations and enough category instances to estimate covariance reliably; combining units across participants could distort the covariance structure on which the geometric quantities depend.

3.9.5 Scope and limitations

The geometric framework characterizes the separability of phoneme and word manifolds under a specified few-shot prototype readout. Its decomposition is diagnostic rather than causal: signal, dimensionality, and signal–noise overlap identify the geometric sources of separability, but they need not correspond to quantities that Whisper independently represents or controls. The prototype classifier is also an analytical readout, not Whisper's learned output rule. Whisper predicts tokens through a trained projection onto its vocabulary, and the present analyses do not establish whether this projection exploits the distinctions revealed by prototype classification. Peaks in prototype separability should therefore not be interpreted as intrinsically optimal processing stages or as evidence that later layers discard information relevant to transcription.

The empirical scope is limited to a single checkpoint, Whisper `large-v3`, and two corpora of read speech. Whether the observed trajectories generalize to smaller Whisper models, self-supervised speech encoders, or systems trained with different objectives remains unknown. LibriSpeech and TIMIT also contain less variation than spontaneous, accented, or noisy speech. Moreover, because Whisper was trained on audio collected from the web, some LibriSpeech recordings may have been present in its training data. The analyses do not require the stimuli to be held out, but possible training-set exposure limits their interpretation as evidence of generalization to novel utterances.

Comparisons between linguistic levels are constrained by differences in stimulus construction and representation extraction. The word and phoneme analyses use different corpora and temporal pooling procedures; their peak locations can therefore be compared along the common encoder depth, but their absolute geometric values are not directly commensurate. Mean pooling further removes the temporal order of frames and may affect units of different durations differently. The analyses also rely on forced word alignments and manual phonetic boundaries. They characterize representations within known intervals but do not address how Whisper discovers those intervals

during recognition.

The decoder analysis has a further interpretive constraint. Representations are measured before the first token of the target word enters the decoder, so they describe the state used to predict that token rather than a representation formed after the complete word has been decoded. Because words may comprise different numbers and identities of subword tokens, decoder geometry may reflect tokenization and orthographic structure in addition to lexical identity. In the sentence condition, the preceding tokens are ground truth, and the manipulation changes both preceding audio and text. The resulting contextual effect therefore cannot be assigned specifically to acoustic context, linguistic prediction, or normal autoregressive decoding.

The framework itself is limited to second-order summaries of Euclidean geometry. These summaries may not capture curvature, multimodality, or higher-order structure within the manifolds, while finite sampling restricts estimation when classification error approaches zero. The measured quantities are invariant to rotations and uniform rescaling, but not to arbitrary invertible transformations of the representation. The results therefore characterize Whisper's representations in their implemented coordinates under the readout considered here, rather than an invariant notion of all information contained at each layer.

Finally, the proposed neural predictions rely on a coarse correspondence between model depth and stages of auditory processing. They specify what neural recordings should reveal if cortical populations reorganize speech variability in the manner observed in Whisper, but they do not establish that the brain uses the same representations or transformations. Whisper thus serves as a source of quantitatively testable hypotheses, not as a validated model of the auditory system.

3.9.6 Outlook

Three extensions follow directly from the unresolved questions. First, audio-only, text-only, and post-token conditions would separate contextual prediction, acoustic evidence, and explicit lexical

input in the decoder. Regressing and perturbing the context-dependent component of the encoder output would test what information it carries and whether the decoder uses it. Second, matching neighboring phonemes across words would distinguish lexical organization of phoneme variability from shared local phonetic context.

The broader priority is to test whether the observed sequence generalizes across models and biological systems. Comparing Whisper with self-supervised speech encoders, or comparing the same encoder before and after transcription fine-tuning, would isolate the roles of architecture and training objective. Framewise analyses could determine how the geometry develops around phoneme and word boundaries without relying exclusively on pooled representations. Neural population recordings could then test whether auditory and language-responsive populations exhibit the same progression from expansion to signal amplification and from phoneme to word separability.

Chapter 2 used the signal-to-noise ratio to compare geometric solutions across architectures; this chapter used the same framework to follow transformations across the depth of one model, across linguistic units, and under sentence context. Together, the two applications show how population geometry can connect category-level performance to the organization of variability within a representation. Chapter 4 considers the implications of this common description across visual reasoning and speech recognition.

3.10 Methods

3.10.1 The Whisper model

All analyses were performed on OpenAI's Whisper [18], an encoder–decoder Transformer trained on weakly supervised multilingual speech, using the `large-v3` checkpoint distributed with release `20250625` of the `openai-whisper` package under PyTorch 2.14. The original Whisper models were trained on 680,000 hours of audio; the `large-v3` checkpoint was trained on a larger corpus and

postdates the published description[99]. The model takes a 30-second audio window at 16 kHz, converts it to a 128-bin log-mel spectrogram at 100 frames per second (128×3000), and passes it through a stem of two one-dimensional convolutional layers with GELU nonlinearities, the second with stride 2, so that the encoder operates on 1500 frames of 20 ms. Sinusoidal positional encodings are added, and 32 Transformer encoder blocks process the sequence, each consisting of pre-normalized self-attention and a two-layer MLP with residual connections, followed by a final layer normalization (`ln_post`). The decoder has 32 blocks of the same width, each applying self-attention over the tokens supplied so far, cross-attention over the encoder output, and an MLP, followed by a final layer normalization (`ln`); the next token is the vocabulary entry whose embedding has the largest inner product with the normalized state. The model dimension is 1280 throughout, with 20 attention heads. All forward passes were run in evaluation mode at 32-bit floating-point precision on a single GPU.

Every decoder input begins with the four-token task prefix `<|startoftranscript|>`, `<|en|>`, `<|transcribe|>`, `<|0.00|>`, which fixes the language and task and opens the first transcript segment at time zero. Representations were extracted with a custom forward pass that supplies the decoder input explicitly, as described below.

3.10.2 Layer indexing and activation extraction

The layer axis $k = 0, \ldots, 66$ is defined in Section 3.3; Table B.2 lists the module read at each index. Index 0 is the encoder input: the output of the second convolution after its GELU, with the sinusoidal positional embedding added. Index $k = 1, \ldots, 32$ is the residual stream returned by `encoder.blocks[`$k-1$`]`, after both the self-attention and the MLP outputs have been added; nothing is read between the two sublayers. Index 33 is the output of `encoder.ln_post`. Index $33 + j$ for $j = 1, \ldots, 32$ is the residual stream returned by `decoder.blocks[`$j-1$`]`, after its three residual additions (self-attention, cross-attention, MLP). Index 66 is the output of `decoder.ln`. The decoder

input, the token embedding plus the learned positional embedding at each position, is also extracted but is not assigned an index (see below).

Every input is a waveform of at most 30 s that is zero-padded in the waveform to the full 30 s window before the log-mel transform, following the package's low-level path (`pad_or_trim`, then `log_mel_spectrogram`). The padded interval therefore enters the model as the log-mel value of digital silence, the floor of the normalized spectrogram. The package's `transcribe` call handles a short clip differently: it computes the spectrogram of the clip alone. It pads the normalized log-mel array with zeros, and a normalized value of zero corresponds to a log energy of -4 rather than to silence. The two conventions give the same encoder frames over the word and different frames after it, and the decoder, which attends to all 1500 frames, is sensitive to the difference (Section 3.9.5). Every input in this chapter, including those of the transcription pass of Section 3.10.4, is padded in the waveform before it reaches the model.

At every encoder index, the readout is a 1500×1280 array over frames and units. The aligner gives a word's onset and offset as integer counts of 10 ms ticks; both are first rounded down to the encoder's 20 ms frame grid, giving boundaries n_s and n_e in ticks, so that the audio cropped for the word is the same in every condition for the sample. The word then occupies encoder frames $n_s/2$ through $n_e/2 - 1$ of the input window, at 50 frames per second, and its representation is the mean over exactly those frames; the rounding trims at most 10 ms at either edge and no other margin is added or removed. In the single-word condition the cropped audio starts the window, so $n_s = 0$; in the in-sentence condition n_s and n_e are the word's times in the sentence. The spectrogram baseline averages the 128 mel channels over the spectrogram frames aligned in time with those encoder frames, n_s through $n_e - 1$ at 100 frames per second, so that baseline and encoder are pooled over the same nominal interval.

At every decoder index, the readout is a $T \times 1280$ array over the T input positions, and the representation is the row at the last position, $T - 1$. In the single-word condition, the input is the four-

token prefix of Section 3.10.1, so $T = 4$ and the position read is that of `<|0.00|>`; in the in-sentence condition, the input is the prefix followed by the tokens of the preceding words (Section 3.10.4), and the position read is that of the last context token. The whole input is presented in one pass, which yields the same states as the package's cached incremental decoding. At the decoder input, the state read depends only on the last input token and its position, since cross-attention to the audio first enters in the first block. In the single-word condition, the input token is the same for every utterance, so the decoder-input representation is a single vector for the whole stimulus set, and no pairwise quantity is defined for it; in the in-sentence condition it varies with the last context word but not with the target. The decoder input is therefore kept off the layer axis and is not plotted; where it is used for the in-sentence condition, it is referred to by name.

3.10.3 Word stimuli and word representations

Words were drawn from the `test-clean` and `train-clean-360` partitions of LibriSpeech [100], a corpus of read English audiobook speech at 16 kHz. Word boundaries were taken from the LibriSpeech Alignments release [101], forced alignments of the reference transcripts produced with the Montreal Forced Aligner [102] and its pretrained LibriSpeech acoustic model; the segmentation is therefore independent of the model under study. Alignments exist for 106,628 of the 106,634 utterances. We dropped the six utterances without one, the 8,671 tokens the aligner could not label, and apostrophe-bearing forms, leaving 50,658 word types. An occurrence was counted as eligible only if the word began at least 5 s after the sentence onset and ended within the sentence's first 30 s. Word types with at least 200 eligible occurrences in distinct sentences formed the candidate pool, and 200 occurrences of each were sampled uniformly without replacement under a fixed seed, at most one from any sentence. We then applied three exclusions: words that no single Whisper vocabulary token decodes (i.e., they can be produced only as a sequence of word-pieces); function words, identified by part-of-speech tag; and a manually specified list of numerals, proper names, and closed-class

words the tagger misreads. The tagger was applied to each word in isolation, without sentence context. Hence, it labels auxiliaries and copulas as main verbs, and the manual list exists to catch what it misses. Appendix B.1.1 gives the pool size, the number removed at each step, and the closed-class words that survive both steps; the final set has 800 word types.

In the single-word condition, we excised the audio between the word's boundaries n_s and n_e (Section 3.10.2) from its sentence and zero-padded it to 30 s, with the word at the start of the window. Representations were read from each layer as described in Section 3.10.2.

3.10.4 Words in sentence context

In the in-sentence condition, the input is the sentence audio from its onset through the target word's offset n_e, zero-padded to 30 s, so the model hears everything that preceded the word and nothing that followed it. The eligibility rule of Section 3.10.3 guarantees that every in-sentence input carries several seconds of preceding speech and fits the window without truncation.

The decoder input for this condition is the four-token prefix followed by the ground-truth words of the sentence up to, but not including, the target word. These words are tokenized with Whisper's tokenizer, each with a leading space, exactly as the model would tokenize that text.

A position-matched control condition presents the isolated word preceded by n_s ticks of silence, so that its onset falls on the same encoder frame as in the in-sentence condition, and the decoder input is the four-token prefix of the single-word condition. In the encoder, the control therefore differs from the in-sentence condition only in the preceding audio, silence in place of speech; in the decoder, it differs in the preceding audio and in the decoder input, since the readout position is the number of preceding tokens and cannot be matched without supplying them. The control is run for both stacks, with the same extraction, geometry, and readout as the two main conditions.

Each input was transcribed with Whisper's default `transcribe` call (English). Whisper transcribes 79.5% of the isolated words and 90.2% of the same words in sentence context correctly.

Accuracy is not a selection rule: Section 3.8 reports all 160,000 occurrences, and the comparison restricted to the 118,041 occurrences transcribed correctly in both conditions is reported as a control in Appendix B.6.1.

3.10.5 Phoneme stimuli and phoneme representations

Phoneme-level analyses drew candidate occurrences from all 6,300 utterances of TIMIT[104], read by 630 speakers of American English at 16 kHz, each with a manual phonetic transcription whose boundaries are given as sample indices. The 61 transcription labels were reduced to 39 speech labels by a mapping derived from that of Lee and Hon[105], tabulated in Appendix B.1.2, with the following departures from their evaluation set. The pause, silence, and epenthetic-silence symbols and the glottal stop were removed rather than pooled into a silence class; /ZH/ was kept separate from /SH/ and /AO/ from /AA/; and the flap was folded into /D/. After reduction, temporally adjacent intervals carrying the same label were merged, so that a stop closure immediately followed by its release formed one interval with the stop's label, while an unmatched closure remained a separate occurrence of that label. Occurrences were pooled across words and speakers, so a phoneme manifold spans the coarticulatory, speaker, and rate variability of the phoneme in connected speech, and the readout tests generalization across occurrences rather than across held-out speakers. Each phoneme was represented by 200 occurrences, the manifold size used for words, sampled as the words were: one occurrence chosen at random in every utterance containing the label, then 200 of those utterances drawn uniformly without replacement, both under the configuration seed. The same 200 were used at every layer and in every analysis. Every label satisfies the rule; the least frequent, /ZH/, has 225 occurrences in 219 utterances, and each label's 200 occurrences come from between 162 and 183 speakers.

Each utterance containing a selected occurrence was presented to the encoder once, whole, at the start of the input window and zero-padded to 30 s, and the residual stream was read at encoder

indices 0 to 33 (Section 3.10.2); every selected occurrence in the utterance was pooled from that pass. Encoder positions lie on a nominal 20 ms grid, one per 320 samples, and the representation of an occurrence with boundaries s and e in samples is the mean over the positions whose nominal bins overlap the interval, $\lfloor s/320 \rfloor$ through $\lceil e/320 \rceil - 1$; the mel baseline applies the same rule with 160 samples per frame. This is looser than the word rule, which keeps only positions wholly inside the word, because phones are short: the median selected phone lasts 82 ms, the median number of pooled positions is five, and some flaps and unreleased stops are shorter than one bin, so the rule guarantees every occurrence at least one position. No decoder representation is defined for a phoneme, since decoder positions index text tokens and have no temporal alignment with phone intervals; the phoneme analyses are encoder-only. Geometry and the prototype readout are computed as for words, over all 39×38 ordered pairs at $m = 5$, the two directions of a pair being treated as one block in every resampling and test (Section 3.10.7).

Phonemes are compared with a set of distinctive features. Consonants are described by manner (stop, affricate, fricative, nasal, lateral, rhotic, glide), place (labial, dental, alveolar, postalveolar, palatal, velar, glottal) and voicing; vowels by height (high, mid, low), backness (front, central, back), rounding, tenseness, and whether the vowel is a diphthong. The feature distance between two phonemes is the number of features in which they differ; consonant–vowel pairs are not assigned a distance and form a class of their own. Every pair of the 39 labels differs in at least one feature, and 53 consonant pairs and 17 vowel pairs differ in exactly one, among them the eight voicing pairs (/P/–/B/, /T/–/D/, /K/–/G/, /CH/–/JH/, /F/–/V/, /TH/–/DH/, /S/–/Z/, /SH/–/ZH/). In matrix figures, the phonemes are ordered by manner class, within a class by place from front to back with the voiceless member of a voicing pair first, and the vowels around the vowel quadrilateral from high front through low to high back, followed by the diphthongs; the same order is used at every layer. The feature table is given with the inventory in Appendix B.1.2.

3.10.6 Manifold geometry and the prototype readout

Geometric quantities follow Chapter 2 exactly and are not redefined here. For a set of representations sharing a word or phoneme label, the manifold centroid, covariance, principal radii, mean squared radius, and participation-ratio dimensionality are given by Eqs. 2.2–2.5; the normalized centroid separation is Eq. 2.6; the signal-to-noise ratio governing discriminability of an ordered pair is Eq. 2.7; and the predicted error is Eq. 2.8. Notation follows Table 2.3.

Each word manifold was estimated from its 200 utterances at a given layer, in the full 1280-dimensional space; unlike Chapter 2, where layers of different width were projected to a common dimensionality (Section 2.11.4), no random projection was applied, because Whisper's width is constant across layers. D is the participation ratio of the manifold radii (Eq. 2.5) and is bounded above by the number of utterances per manifold.

Empirical separability was measured with the m-shot prototype classifier of Eq. A.4 with $m = 5$ throughout. For each ordered pair of words, m utterances of each word were drawn at random and averaged to form two prototypes, and each remaining utterance of the first word was assigned to the nearer prototype under Euclidean distance; the error is the fraction assigned to the wrong word, averaged over 200 random draws of the prototypes. The corresponding theoretical prediction is $\varepsilon = H(\mathrm{SNR}_m)$ with $H(x) = \frac{1}{2}\,\mathrm{erfc}(x/\sqrt{2})$, where SNR_m is the m-shot prototype SNR of Eq. A.4: relative to the one-shot SNR of Eq. 2.7, averaging m examples into the prototype reduces the bias term, the dimensionality term, and the overlap of the centroid separation with the competing manifold's axes by the factor m, while the overlap with the test manifold's axes is unchanged. All 800×799 ordered pairs were evaluated. Wherever a quantity is reported "per layer" it is the mean over these pairs; the error bands are $\pm$ one standard error of the mean across the 800 words, each word's value being the mean over its 799 ordered pairs.

The estimator carries two further variance terms that Eq. A.4 does not display; Appendix B.3

derives the full expression.

The linear classifier of Section 3.4.1 is a single linear layer from the 1280 units to the word labels, trained with the cross-entropy loss on all 800 words jointly. Training uses Adam at a learning rate of 10^{-3} with weight decay 10^{-4}, mini-batches of 128 occurrences, and 200 passes over the training set, on representations that are not centered or rescaled. Accuracy is estimated by five-fold cross-validation over the 200 occurrences of each word, with each fold holding out 40 occurrences of every word. It is the fraction of held-out occurrences assigned their correct label. The same classifier is trained on the 39 phoneme labels in Section 3.7; its margin, reported in Appendix B.5.4, is the mean over correctly classified held-out occurrences of the difference between the score of the correct label and the largest score of any other label.

3.10.7 Linguistic feature measures and matched subsampling

Three measures of the difference between two words were computed on word types. *Duration difference* is the absolute difference between the two words' mean durations, each mean over the word's 200 occurrences (Section 3.10.3). *Phoneme-sequence distance* is the Levenshtein edit distance between the two words' dictionary pronunciations, the first pronunciation listed for the lower-cased word in the CMU Pronouncing Dictionary distributed with NLTK[106], with stress markers removed; all 800 words have an entry. *Semantic distance* is one minus the cosine similarity between the two words' 300-dimensional word2vec embeddings trained on Google News[107], loaded through `gensim` as `word2vec-google-news-300`; every word of the set has an entry under its lower-case form. A second semantic measure, used in Appendix B.4, is the same distance between type-level embeddings from `bert-base-uncased`[108]: each word was presented alone between the `[CLS]` and `[SEP]` tokens, and its embedding is the final-layer hidden state of the corresponding WordPiece token; every word of the set is a single WordPiece token.

Four further quantities serve as controls. The *phoneme-count difference* is the absolute difference

in the number of phonemes of the two pronunciations. The *input acoustic distance* is the signal term of Eq. 2.6 between the two words' manifolds in the 128-dimensional log-mel representation, computed as in Section 3.10.6 over the same occurrences. The pair's *mean duration* is the mean of the two words' mean durations. *Word frequency* is the base-10 logarithm of the word's total number of occurrences in the aligned corpus of Section 3.10.3, counted before selecting any occurrence, and enters as the absolute difference between, and the mean of, the two words' log frequencies. The *grammatical class* of a word, used in Appendix B.4, is noun, verb, adjective, or adverb according to the part-of-speech tag assigned to the word in isolation (Section 3.10.3); 12 words with another tag are excluded from the class comparison. All measures are symmetric in the two words. The signal-to-noise ratio is directed, so each selected unordered pair enters as both ordered pair; the two form one block in every resampling, and at each layer we average the ratio and its tagged version over all ordered pairs in the group.

For each target measure, unordered word pairs were divided into a large-difference and a small-difference group, with the controls balanced by stratification: the other two measures, the phoneme-count difference and the input acoustic distance for every target; in addition, the pair's mean duration for the duration target, and the pair's difference in and mean of log frequency for the semantic targets. When used as controls, the three difference measures and the log-frequency difference were each divided into six equal-width bins spanning their ranges across all word pairs, closed on the left with the upper endpoint in the last bin; the phoneme-count difference was taken at its exact value, with differences of four or more as one stratum; the input acoustic distance, the pair mean duration and the pair mean log frequency were each cut into quartiles over all pairs. Each pair was assigned to the cell given by its stratum indices. In each cell holding 100 pairs or more, the largest 5% of pairs by target value, rounded down, formed the large-difference group and an equal number with the smallest values the small-difference group; ties at a cutoff were broken at random. Cells with fewer than 100 pairs, or in which the smallest selected large value did not exceed the largest selected small value,

were excluded; no cell containing at least 100 pairs failed the latter criterion. Of the possible cells, 391 of 2,880 (duration), 156 of 720 (phoneme-sequence distance) and 623 of 17,280 (semantic distance) were retained. Each matched set of word pairs was constructed once from the type-level measures and then used unchanged to evaluate the layer-wise geometry in the single-word and, where both conditions were analyzed, the in-sentence condition.

Residual imbalance is reported for each control as the standardized mean difference between groups over the selected unordered pairs: the large-group mean minus the small-group mean divided by the pooled within-group standard deviation; target separation is the difference between the two groups' mean target values. Both are given with Figures 3.7 and 3.8. Three further targets are matched in the same way for Appendix B.4: the BERT semantic distance, with the controls of the word2vec comparison; the input acoustic distance, on the three measures and the phoneme-count difference; and grammatical class, a binary target with the controls of the word2vec comparison, for which, within each cell holding 100 pairs or more, all pairs from the smaller category and an equal number sampled at random from the larger were retained.

Uncertainty was estimated by bootstrap over word types with the matched groups held fixed: the bins, the retained cells, and the selected tails are those of the full sample. In each of $B = 10{,}000$ replicates, the 800 word types were drawn with replacement; a pair whose words were drawn m_i and m_j times entered with weight $m_i m_j$, pairs formed from two copies of the same word were excluded, and the two ordered pairs of an unordered pair carried the same weight. Per-layer group means were recomputed with these weights. The bands are the pointwise 2.5th and 97.5th percentiles of the replicates. We tested the difference between the two groups' ratio curves using the same replicates: in each replicate, we computed the large-minus-small difference at every layer, subtracted the observed difference, and recorded the largest absolute value of this centered difference over the 67 layers. The 95th percentile of these maxima is the half-width of a simultaneous 95% interval around the observed difference at every layer, and layers whose interval excludes zero are identified. The in-

terval was constructed separately for each target measure and input condition, controlling across the 67 layers within that comparison. The inference is applied to the ratio only; its terms are reported descriptively.

4

Conclusion

4.1 A common geometric account of few-shot separability

This dissertation applies a geometric theory of few-shot classification, originally developed for perceptual object categories [15], to two new domains: abstract relational rules in visual reasoning networks, and linguistic units—phonemes and words—in Whisper, a speech-recognition model [18]. In both studies, few-shot classification is evaluated by estimating a prototype for each category from a

small labeled sample and assigning held-out examples to the nearest prototype.

The central empirical finding is that this theoretical prediction holds in both domains: the accuracy of the prototype classifier is predicted by the geometry of the category representations. The signal-to-noise ratio of Eq. 2.7 combines a small set of interpretable geometric quantities to predict classification error in both the visual and speech studies (Sections 2.5.5 and 3.4.3). This relationship provides a common framework for explaining how different representations support few-shot classification, even when their geometry changes in different ways across network stages.

The two studies provide complementary forms of evidence. Chapter 2 uses synthetic stimuli and fully specified rules to isolate compositional generalization: each held-out rule shares a relation and an attribute with training rules but not their combination. Training four architectures on the same data and objective permits controlled comparisons across architectures, layers, and learning, while direct optimization of the theoretical signal-to-noise ratio tests whether the framework can guide representation learning. Chapter 3 applies the same framework, without modification, to a large pretrained encoder–decoder model operating on natural speech. It follows phoneme and word manifolds through depth and examines how the geometry of identical words changes when they are presented alone or after their preceding sentence. Agreement across systems that differ in architecture, input, objective, category structure, and training history supports the generality of the geometric description without implying that they implement a common mechanism.

The dissertation thereby extends manifold theory beyond the perceptual object categories for which it was initially developed[15]. It applies the framework to abstract relational rules, including novel combinations withheld during visual training, and to linguistic units represented throughout an autoregressive encoder–decoder system. It also traces geometry across learning, network depth, linguistic scale, and sentence context, revealing transformations that cannot be inferred from endpoint accuracy alone. Finally, by using the signal-to-noise ratio as a training objective, it moves from diagnosing representational geometry to testing whether that geometry can serve as a target for im-

proving few-shot generalization.

4.2 Multiple geometric routes to separability

The central result across the two studies is that improved separability does not identify a unique representational transformation. Categories can become easier to distinguish because their centroids move farther apart, because their internal variability changes, or because that variability becomes less aligned with the directions separating them. Under the prototype readout studied here, all of these transformations affect error.

The visual study demonstrates this multiplicity under controlled conditions. Architectures trained on the same data and objective develop rule separability through different layerwise trajectories: gradual in the Vision Transformer, abrupt at the first multilayer-perceptron stage of the Scattering Compositional Learner, and established early in the Wild Relation Network (Section 2.6). In SCL, the improvement reflects a structured tradeoff in which signal increases as held-out rule manifolds compress and become aligned with the subspace spanned by the training rules (Section 2.7). Compositional generalization therefore does not arise from separation alone. Novel relation–attribute combinations become accessible as their variability is organized relative to structure shared with previously learned rules.

The speech study reveals a complementary solution. The first gain in word separability in the Whisper encoder occurs without increased signal or reduced variability. Instead, within-word variability spreads across more dimensions and becomes less aligned with word-separating directions. Signal subsequently increases as dimensionality falls, carrying separability toward its encoder peak; in the decoder, improvement is dominated by amplification without a comparable sustained expansion (Section 3.5). Improved representations therefore need not suppress within-category variation: variability can persist, or even expand, if weakly aligned with category-separating directions.

Which geometric transformation improves separability depends on the categories being distinguished. Phonemes and words become separable at different depths and through different transformations within the same network: whereas the initial expansion of word manifolds improves their separability, phoneme manifolds expand while losing signal, leaving separability nearly unchanged; phoneme separability later improves through signal amplification and peaks nine layers before that of words (Section 3.7). Sentence context produces another dissociation. At the encoder output, preceding speech introduces low-dimensional variability aligned with word separation and sharply reduces separability. In the late decoder, contextual variability is high-dimensional and weakly aligned, and words become more separable after their sentence than after position-matched silence despite having weaker signal (Section 3.8).

These geometric differences need not be visible in aggregate decoding accuracy. Across decoder layers 57–66, linear-probe accuracy remains between approximately 0.94 and 0.96 while prototype error changes by roughly eightfold. The phoneme analyses likewise show that probe performance and few-shot separability can follow different trajectories. Neither readout is uniquely correct: a fitted linear probe and a few-shot prototype classifier ask different questions about what a representation makes accessible. Performance under one readout therefore cannot, by itself, characterize how category information is organized.

Taken together, the results replace a single-axis account of representational improvement with a multivariate one. Signal amplification, expansion, compression, and changes in signal–noise alignment provide distinct and interacting routes to separability, and their contributions depend on the architecture, processing stage, category, and input condition. The recurrence of tradeoffs between signal and dimensionality in both studies does not establish a common cause, but it raises a broader question: whether such tradeoffs are a general consequence of learning to preserve category distinctions while accommodating variation.

These routes ground specific predictions for both, developed in the remainder of the chapter.

Most immediately, the multiplicity of routes predicts that few-shot performance—in models or in neural populations—can improve without any reduction in total within-category variability, provided that variability reorganizes away from category-separating directions. For neuroscience, Section 4.3 derives falsifiable population-level predictions from each study, and Section 4.4 commits to the single most urgent recording experiment. For artificial intelligence, Section 4.4 names the open challenges in visual reasoning and speech processing and commits to algorithmic uses of the framework, building on the demonstration that the signal-to-noise ratio can serve directly as a training objective (Section 2.8).

4.3 From geometry to behavior and neural populations

The framework provides a common analysis for artificial and biological population representations. Each term can, in principle, be estimated from recordings containing repeated instances of a category across variations in stimulus and condition, as in previous analyses of object manifolds in cortex [14,15]. Populations with similar decoding accuracy may nevertheless differ because one increases centroid separation while another reduces, redistributes, or reorients within-category variability. Distinguishing these transformations would yield different accounts of the population computation underlying the same behavioral improvement. The method is therefore transferable to neural data, although the present results do not establish that the brain uses the transformations observed in either model.

The visual study provides a specific link from geometry to behavior. SCL, the architecture with the strongest compositional generalization, also most closely reproduces the ordering of relational difficulty in human errors and response times (Section 2.4). Within SCL, differences in difficulty across relations are carried primarily by the signal between rule manifolds (Section 2.5.5). If the behavioral correspondence reflects analogous representational constraints, rule pairs that humans

find more difficult should exhibit smaller centroid separation relative to within-rule variability in prefrontal or parietal populations representing relational structure. Reliable difficulty differences explained primarily by dimensionality or signal–noise overlap, despite comparable normalized separation, would challenge this proposed correspondence with SCL.

The visual study also motivates a prediction about how learning supports relational transfer. In the visual models, representations of held-out rules increasingly align with a low-dimensional subspace defined by training rules (Section 2.7). If biological learning produces a similar organization, improved one-shot generalization to unfamiliar relation–attribute combinations should be accompanied by increased alignment of their population responses with the subspace established by familiar rules. This prediction could be tested by estimating the familiar-rule subspace at successive stages of training and evaluating its alignment with held-out-rule responses using independent data and control subspaces of matched dimensionality. Reliable improvements in transfer without increased alignment would challenge this proposed learning trajectory. A more specific hypothesis is that alignment is accompanied by increasing normalized centroid separation and decreasing within-rule dimensionality. Consistent increases in both signal and dimensionality would challenge that compression trajectory, while remaining compatible with reuse of a shared relational subspace.

Distances, radii, and dimensionalities can be measured from any population response; the empirical content of the theory lies in the equation that predicts prototype error from them, with no parameters fitted to the errors. Systematic discrepancies beyond estimation uncertainty and expected approximation error would challenge this account. In particular, category pairs matched on all geometric terms entering the equation but exhibiting reliably different prototype errors would demonstrate that the summary omits relevant structure or that the approximation fails in those conditions. Related work on linear classification capacity shows that correlations between manifold centroids and axes must also be accounted for [109]. Comparing predicted errors with behavioral errors tests the additional hypothesis that a prototype readout explains biological few-shot learning; a

behavioral mismatch alone would not invalidate the equation for the specified readout.

The speech results motivate predictions about the ordering of population transformations. If biological speech processing recapitulates the computational sequence observed in Whisper, early gains in word discriminability should arise primarily through an increase in dimensionality and a reduction in signal–noise overlap, with little change in signal. Later gains should instead be dominated by increasing signal as dimensionality falls. Phoneme variability should also expand early, but with a concurrent loss of signal, and phoneme discriminability should reach its maximum before word discriminability. This ordering would be contradicted if early improvements were driven primarily by signal, later improvements occurred without amplification, or word discriminability peaked before phoneme discriminability.

Sentence context yields a further prediction about the organization of variability. If the ordering observed in Whisper generalizes, representations at earlier processing times should preserve the geometry of the acoustic word more strongly, whereas later representations should incorporate contextual variation in a higher-dimensional form that is less aligned with word identity. This prediction concerns changes across processing time or populations, not a direct correspondence between model layers and cortical areas. It would be contradicted if context instead produced strong early disruption followed by increasingly low-dimensional variability aligned with the directions separating words.

4.4 Boundaries of the account and the experiments they motivate

The theory is a second-order approximation to the error of a prototype readout, representing each category manifold through its centroid and covariance. Curvature, multimodality, local nonlinear structure, and higher-order statistics remain outside this description. The prototype readout is also an analytical probe rather than the models' learned decision rule. Peaks in prototype separability

therefore need not identify optimal processing stages, and subsequent declines do not imply that later layers discard information required by the trained model.

The decomposition identifies how signal, bias, dimensionality, and signal–noise overlap contribute mathematically to changes in separability, but it does not explain why a particular architecture or objective produces those changes. Optimizing the theoretical signal-to-noise ratio shows that the framework can guide representation learning and improve held-out generalization; it does not establish that ordinary training reaches its solution through the same route or that any single geometric term causally mediates the improvement. The empirical evidence is also limited to controlled visual stimuli and rule structures, one Whisper checkpoint, two read-speech corpora, externally supplied phoneme and word boundaries, and a fixed padding convention. Agreement across the two studies supports the portability of the description, but their differences in architecture, input, objective, and training history prevent attribution to a common cause.

Several model-side experiments could narrow these uncertainties. Relating prototype separability to the models' trained output projections would test whether the distinctions exposed by the geometric analysis contribute to their native computations. Measuring the same speech model before and after transcription fine-tuning would reveal which transformations arise with task learning, while comparisons among matched encoders trained with and without the signal-to-noise objective would test whether deliberate geometric changes produce the predicted changes in separability. Applying the analysis across model sizes, training objectives, and speech domains would separately establish which trajectories generalize beyond the systems examined here.

Neural experiments would need to distinguish two questions: whether the geometric approximation predicts prototype error in neural populations, and whether the transformations observed in SCL or Whisper also occur in the brain. One promising priority would be a controlled experiment on relational generalization in prefrontal cortex, motivated by evidence for abstract rule coding and population geometries supporting generalization across task conditions[110,12]. Assuming prior task

training and an established recording preparation, macaques could judge whether two successively presented objects share a cued feature, such as shape. Irrelevant color, size, and position could vary independently, with sessions interleaving restricted and broad variation and including previously unseen feature combinations. Responses measured after the second object and before a randomized response mapping is revealed would allow relational representations to be examined separately from preparation of a particular movement.

Each relation would define a manifold of responses to its different visual realizations. Within each variation condition and prespecified response window, centroids and covariances estimated from one set of trials would be used to predict one-shot and five-shot prototype errors on independent responses through Eq. 2.7, without fitting parameters to classification errors. Prototype examples and test responses would be sampled from the same predefined stimulus distribution, and trial and neuron subsampling would assess the reliability of the estimates. Systematic discrepancies beyond estimation uncertainty and expected approximation error would challenge the adequacy of the second-order account for these populations, under the falsification conditions discussed in Section 4.3. Comparisons with Gaussian surrogate responses having matched centroids and covariances could help assess whether omitted higher-order structure contributes to those discrepancies.

The same experiment could examine whether successful generalization under broader irrelevant variation is associated with greater centroid separation, reduced variability, or changes in its dimensionality and orientation. These alternatives would test the neural relevance of the representational routes identified in the model studies. Relating predicted prototype errors to behavioral errors on novel combinations would address the additional hypothesis that the measured representation and readout help explain biological generalization. A behavioral mismatch would weaken that interpretation without necessarily challenging the prediction of neural prototype error. This experiment would be particularly informative because controlled stimulus variation, population geometry, and behavior could be examined within the same recording budget.

These experiments could test both the geometric theory and the neural relevance of the model findings. A direct test of the theory would ask whether second-order geometry estimated from one set of population responses predicts prototype classification error on independent responses across category pairs, subjecting the account to the falsification conditions set out in Section 4.3; systematic failure in neural data would require revising or abandoning the approximation for those populations. Testing whether neural populations undergo the geometric transformations observed in SCL or Whisper would instead assess the correspondence between those models and the brain. Different neural trajectories would challenge that correspondence without necessarily refuting the geometric theory.

The framework also carries algorithmic implications for artificial systems. In visual reasoning, the central challenge is compositional generalization: networks that master familiar relation–attribute combinations degrade sharply on novel ones (Section 2.3.3), and closing this gap by enlarging training data is sample-inefficient. This dissertation already demonstrates a direct response: optimizing the theoretical signal-to-noise ratio as a training objective improves held-out compositional generalization (Section 2.8). The committed proposal is therefore to incorporate the signal-to-noise ratio of Eq. 2.7 as an auxiliary loss on intermediate representations during ordinary task training—penalizing within-rule variability aligned with rule-separating directions and encouraging alignment of novel combinations with the subspace spanned by trained rules—since that alignment is the geometry that successful generalization in these models in fact exhibits (Section 2.7).

In speech processing, the corresponding challenges are word representations that remain stable across speakers, acoustics, and preceding context, and adaptation to new vocabulary or domains from few labeled examples. For few-shot adaptation, the geometric analysis supplies a layer-selection criterion: prototype separability varies by nearly an order of magnitude across late decoder layers whose probe accuracy is indistinguishable (Section 3.4.2), so few-shot readouts for new words should be attached at the geometrically identified layer rather than at the model output. For con-

text robustness, the committed objective is a fine-tuning regularizer that keeps contextual variability high-dimensional and weakly aligned with word-separating directions—the organization through which the late decoder itself makes words more separable after their sentences than after silence (Section 3.8)—rather than suppressing contextual variation outright. In both domains, the framework thus converts measured geometry from a diagnostic into an explicit design target, specifying which quantities to optimize and where in the network to intervene.

Taken together, this dissertation extends a quantitative account of finite-sample separability from perceptual object manifolds to abstract relational rules and linguistic units. Its central contribution is to show that generalization depends not only on whether categories are separated, but on how variability is organized around the directions that separate them. Across visual reasoning and speech, similar levels of separability arise through different combinations of amplification, expansion, compression, and alignment. The framework makes these solutions commensurable without treating them as equivalent, identifies contributions that aggregate accuracy alone cannot resolve, and converts proposed links between population geometry, computation, and behavior into explicit empirical tests.

A

Supporting Material for Chapter 2

This appendix collects the supporting analyses for Chapter 2.

A.1 The *SimplifiedRPM* dataset

SimplifiedRPM is a row-based abstraction of Raven's Progressive Matrices (RPM) in which each example consists of a single row of three panels governed by one abstract relational rule (Fig. 2.2a).

A.1.1 Abstract relations and rule definitions

Each relational rule in *SimplifiedRPM* is defined by applying an abstract relation to a specific object attribute. The dataset includes ten abstract relations spanning three abstract relation types: Progression (constant, progression ± 1, progression ± 2), Arithmetic (addition, subtraction), and Logical (AND, OR, XOR). Each relation specifies how attribute index values assigned to objects evolve across the three panels of a row. Formal symbolic definitions and illustrative examples for all relations are provided in Table 2.1.

A.1.2 Object attributes and visual instantiation

Each relational rule in the *SimplifiedRPM* dataset is applied to one of four object attributes: shape, size, shade, or number/position (Fig. 2.2b). Number and position are grouped as a single attribute to avoid rule ambiguity. For example, a constant-position rule inherently fixes the number of objects, enforcing a constant-number relation and violating the requirement that each row instantiate a single unique relational rule.

To ensure consistency with prior benchmarks, we follow the object definitions and attribute specifications used in the PGM dataset[30]. Shape is selected from seven geometric forms: circle, triangle, square, pentagon, hexagon, octagon, and star. Object size is defined by the height of the object's bounding box and is discretized into ten evenly spaced levels ranging from 20 to 40 pixels. Shade is represented as grayscale intensity and discretized into 10 levels, ranging from white $[230, 230, 230]$ to black $[0, 0, 0]$ in RGB space.

The number attribute specifies the total number of objects present in a panel, ranging from 1 to 9. Object position specifies the spatial arrangement of objects within a panel. Panels are rendered at a fixed resolution of 160×160 pixels, with each object occupying a maximum bounding box of 40×40 pixels. Objects are placed at one of nine predefined spatial locations arranged in a 3×3 grid,

with centroid coordinates:

$$\{(20, 20), (20, 60), (20, 100), (60, 20), (60, 60), (60, 100), (100, 20), (100, 60), (100, 100)\}.$$

Each dataset instance, therefore, has dimensions $3 \times 160 \times 160$, with the three panels presented horizontally.

A.1.3 Dataset generation procedure

Rows in the *SimplifiedRPM* dataset are generated using a structured sampling procedure to ensure that each instance instantiates a single, unambiguous relational rule. For a given rule, attribute values are first sampled such that the target relation is satisfied across the three panels. Values for all remaining, task-irrelevant attributes are then sampled independently at random. To prevent unintended relational regularities, rows are rejected and resampled if any non-target attribute inadvertently satisfies a different relational rule.

Once attribute values are finalized, objects are rendered with the specified attributes and placed at their designated spatial locations to form the three-panel row. This procedure enforces the one-rule-per-row constraint while ensuring substantial visual variability across instances of the same rule through randomization of task-irrelevant attributes.

A.1.4 Dataset size and splits

For each of the 40 relational rules, 10,000 training rows, 1,000 validation rows, and 1,000 test rows were generated, yielding a total of 12,000 rows per rule and 480,000 rows overall. In each experiment, models are trained on a subset of 35 rules and evaluated on the remaining 5 held-out rules, with no overlap between training and test rules. Fig. 2.2b provides an example of this rule split. Rule splits are sampled randomly, subject to the constraint that each abstract relation type appears

at most once in the held-out set.

A.1.5 Low-level visual statistics and shortcut audit

To evaluate whether models could solve the *SimplifiedRPM* task by exploiting low-level image statistics rather than learning the intended abstract relational structure, we conducted a systematic audit of pixel-level visual features. For each panel, we measured foreground occupancy, edge density, mean luminance, and luminance contrast. We defined foreground occupancy as the proportion of pixels belonging to objects rather than background, edge density as the fraction of pixels identified as edges by a Canny detector, mean luminance as the average pixel intensity, and contrast as the variance of pixel intensities. Each statistic was computed separately for the three panels of a row, yielding a three-dimensional feature vector per statistic.

To test whether low-level visual features provide separable or predictive signals for relational rules, we analyzed both statistical separation and feature-only classification performance. Multivariate analyses of variance (MANOVA) across the 40 rules using 1,000 test rows per rule revealed statistically significant effects of rule identity for all individual features (foreground occupancy, edge density, mean luminance, and luminance contrast); however, effect sizes were uniformly small, with rule identity explaining only 0.6–3% of the total variance. Concatenating all four features into a 12-dimensional panel-wise feature space did not increase separability, with rule identity accounting for only about 3% of the variance. Consistent with these results, multinomial logistic regression classifiers trained using only low-level features performed near chance, with classification errors of 0.89–0.94 (chance = 0.975) for individual features and 0.88 when all features were combined, far worse than the performance of the models evaluated in Chapter 2. Together, these analyses indicate that low-level visual statistics do not reliably predict which relational rule is present.

A feature-only matching baseline further evaluated whether low-level visual statistics could support the relational matching structure of the *SimplifiedRPM* task. For each trial, Euclidean dis-

tances were computed between the z-score–normalized feature vectors of the sample row and the two candidate choice rows, and the closer match was selected. This feature-only baseline achieved near-chance accuracy (approximately 0.50). It yielded a mean generalization error of 0.44 ± 0.08, substantially worse than even the least generalizing image-based model evaluated in this study.

Together, these analyses indicate that low-level visual statistics in *SimplifiedRPM* do not reliably encode meaningful information about relational rule identity. Successful performance on the task therefore requires learning the intended abstract relational structure rather than exploiting superficial visual cues.

A.1.6 Transfer under renderer shifts

This work studies representation geometry for relational reasoning in the controlled setting of the *SimplifiedRPM* dataset. To ensure that the geometric findings are not constrained by the specific renderer used in *SimplifiedRPM*, zero-shot transfer is evaluated under renderer shifts that substantially alter low-level visual statistics, including texture, shade, and shape filling (Fig. A.1). This analysis examines whether the relationship between representation geometry and relational generalization persists when visual appearance changes while relational structure is held fixed. For each target renderer, models are evaluated across five different held-out rule splits. Quantitative results across all renderer shifts are summarized in Table A.1.

Texture. In the original *SimplifiedRPM* renderer, objects are depicted as filled shapes with uniform intensity, resulting in minimal internal texture. To introduce controlled texture perturbation while preserving relational structure, we modify the object fill by adding a sinusoidal intensity mod-

ulation within each object's mask. For an image $I(x, y)$, the textured version $\tilde{I}(x, y)$ is defined as

$$\tilde{I}(x,y) = \begin{cases} I(x,y) + a \sin\left(2\pi f \frac{y}{H}\right), & (x,y) \in \mathcal{M}, \\ I(x,y), & (x,y) \notin \mathcal{M}, \end{cases} \tag{A.1}$$

where $\mathcal{M}$ denotes the object mask, H is the image height, f controls the spatial frequency of the texture pattern, and a controls the modulation amplitude. By varying $f \in \{6, 10\}$ and the amplitude a (Table A.1), a family of renderer variants is obtained that systematically increases internal texture while keeping object boundaries, spatial layouts, and relational rules fixed.

Color palette. In the original *SimplifiedRPM* renderer, object fills are drawn from a fixed discrete grayscale palette. To introduce a controlled appearance shift while preserving relational structure, we remap the palette to object fill intensities. Given original fill values $\{v_i\}$ and background intensity b, a new palette is defined

$$v_i' = b + \gamma(v_i - b), \tag{A.2}$$

where γ controls a global contrast change relative to the background. This transformation preserves the relative ordering of fill levels. All object pixels are recolored by mapping each original fill level to its corresponding remapped value, while background pixels remain unchanged. Varying $\gamma \in \{0.5, 0.8, 1.2, 1.5\}$ generates a family of renderer variants with altered color statistics.

Object outlines vs. filled shapes. In the *SimplifiedRPM* dataset, objects are represented as filled shapes. An outline-based renderer replaces each object with its boundary representation. Specifically, we retain only boundary pixels to form an outline, set all interior pixels to the background value, and keep outline pixels at their original intensity. By varying the outline width (w) in pixel units ($w \in \{3, 5, 7\}$), a family of renderer variants is obtained that alters object appearance

while preserving relational structure.

Table A.1: Generalization error under different renderer shifts. Values are mean ± s.e.m. across five held-out rule splits. CE, cross-entropy training; SNR-loss, training with the SNR objective of Section 2.8; a and f, amplitude and spatial frequency of the sinusoidal texture; γ, contrast of the remapped palette relative to the background; w, outline width in pixels.

Texture-based renderer shifts				
Model	**Freq.**	$a = 0.2$	$a = 0.5$	$a = 0.8$
CE	$f = 6$	0.186 ± 0.028	0.270 ± 0.023	0.232 ± 0.021
	$f = 10$	0.152 ± 0.033	0.216 ± 0.035	0.261 ± 0.036
SNR-loss	$f = 6$	0.145 ± 0.016	0.264 ± 0.017	0.321 ± 0.013
	$f = 10$	0.151 ± 0.026	0.285 ± 0.018	0.367 ± 0.013

Color-palette renderer shifts				
Model	$\gamma = 1.5$	$\gamma = 1.2$	$\gamma = 0.8$	$\gamma = 0.5$
CE	0.316 ± 0.030	0.254 ± 0.028	0.156 ± 0.028	0.173 ± 0.028
SNR-loss	0.339 ± 0.012	0.233 ± 0.011	0.134 ± 0.009	0.127 ± 0.016

Outline-based renderer shifts			
Model	$w = 3$	$w = 5$	$w = 7$
CE	0.412 ± 0.018	0.389 ± 0.020	0.354 ± 0.033
SNR-loss	0.448 ± 0.010	0.434 ± 0.011	0.408 ± 0.015

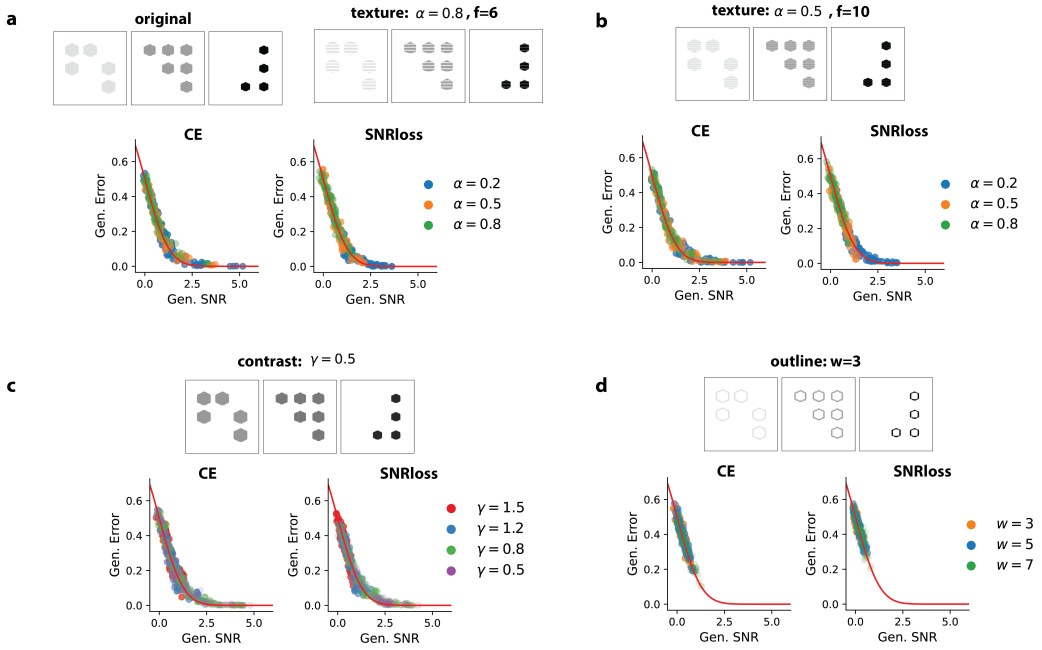


Figure A.1: Geometry predicts relational generalization under renderer shifts. (a) Example rows rendered using the original *SimplifiedRPM* renderer and a texture-based variant with sinusoidal modulation of object fill ($a = 0.8, f = 6$). (b) Texture-based renderer with higher spatial frequency ($a = 0.5, f = 10$). (c) Color-palette remapping with reduced contrast ($\gamma = 0.5$). (d) Outline-based renderer with boundary-only object representations (outline width $w = 3$). For each renderer variant, panels below show generalization error as a function of geometric signal-to-noise ratio (Gen SNR) for models trained with cross-entropy (CE) or SNR-loss. Each point corresponds to a held-out rule pair evaluated zero-shot; colors denote renderer parameters. Red curves indicate the theoretical SNR–error relationship. Across all renderer shifts, Gen SNR remains predictive of generalization error, despite substantial changes in low-level visual statistics.

A.2 Model architectures and training

A.2.1 Model architectures

Four neural network architectures commonly used for visual perception and relational reasoning tasks were evaluated: ResNet-50, Wild Relation Network (WReN), Vision Transformer (ViT), and Scattering Compositional Learner (SCL).

ResNet-50 follows the standard architecture described by [53], consisting of an initial convolutional layer, four groups of residual blocks, global average pooling, and a fully connected classification head (Table A.4). The Wild Relation Network (WReN) processes each panel with a shared convolutional block, constructs pairwise relational representations across panels, and aggregates them through a multi-layer perceptron for classification (Table A.3). The Vision Transformer (ViT) model embeds input patches into token representations and processes them through four stacked Transformer encoder blocks with multi-headed self-attention (Table A.2).

The Scattering Compositional Learner (SCL) applies a shared convolutional block to each panel, followed by a hierarchy of one-dimensional convolutional layers operating across panels and a final multi-layer perceptron (Table A.5). Fig. 2.8 shows an overview of the SCL architecture and its panel-wise processing pipeline. The second MLP layer (MLP2), the penultimate layer of the network and the layer immediately before the classification head, is the relational representation used throughout the *SimplifiedRPM* analyses.

A.2.2 Hyperparameter tuning

Hyperparameters for WReN, ViT, and SCL were selected using a systematic grid search over architectural parameters, with model selection based on minimizing classification error on validation data from training rules. For WReN, the search varied the depth and hidden dimension of both

Table A.2: Vision Transformer (ViT) architecture.

Stage	Module type	Dimension
Inputs	Patch embedding + positional encoding	28×96
Attn1	Transformer encoder block	96
Attn2	Transformer encoder block	96
Attn3	Transformer encoder block	96
Attn4	Transformer encoder block	96
Output	Classification head	35

Table A.3: Wild Relation Network (WReN) architecture.

Stage	Module type	Dimension
Input row	Three image panels	$3 \times 160 \times 160$
CNN	Shared convolutional block	3×80
Projection	Linear projection	3×256
Pairwise	Pairwise relational module (2 layers, 512 hidden)	256
MLP1	Fully connected layer	256
MLP2	Fully connected layer	256
MLP3	Fully connected layer	256
Output	Fully connected layer	35

Table A.4: ResNet-50 architecture.

Stage	Module type	Dimension
Input	Image	$3 \times 160 \times 160$
Conv1	Convolutional layer	$64 \times 80 \times 80$
Block 1	Residual block group	$256 \times 40 \times 40$
Block 2	Residual block group	$512 \times 20 \times 20$
Block 3	Residual block group	$1024 \times 10 \times 10$
Block 4	Residual block group	$2048 \times 5 \times 5$
avgpool	Global average pooling	2048
FC	Fully connected	35

the pairwise relational module ({1–4} layers; {128, 256, 512} units) and the classification MLP ({1–4} layers; {64, 128, 256} units), selecting a configuration with a 2-layer, 512-dimensional relational module and a 3-layer, 256-dimensional classification MLP. For ViT, the search covered the number of encoder layers ({4, 8, 12}), attention heads ({4, 8, 12, 16}), and hidden dimensions ({48, 96, 192, 384, 768}), selecting a 4-layer model with 12 heads and a hidden dimension of 96. For SCL, the search varied the kernel size of the 1-D convolutional layers ({1, 2, 4, 8}), the number of output channels ({16, 32}, {32, 64}, and {5, 10} for successive layers), and the MLP hidden dimension ({400, 1000}), selecting a configuration with kernel size 1, channel sizes {64, 32, 5}, and an MLP hidden dimension of 400. ResNet-50 follows a standard architecture and did not require hyperparameter tuning.

A.2.3 Training

All models were trained using supervised learning with a cross-entropy loss. Training was performed using stochastic gradient descent (SGD) with a learning rate of 0.1, momentum of 0.9, weight decay of 5×10^{-4}, and a batch size of 512. To improve training stability, a learning rate warmup was applied for the first two epochs, followed by a plateau-based scheduler that reduced the learning rate by a factor of 0.2 when the validation loss plateaued. Training proceeded until classification error on 1,000 validation examples per training rule stabilized. The final training durations were 100 epochs for ResNet-50, 60 epochs for WReN, 200 epochs for ViT, and 150 epochs for SCL.

Table A.6 summarizes training-time computational cost, measured by wall-clock time, per-step latency, throughput, and peak GPU memory under identical settings. All models trained stably across five random seeds, with no numerical instabilities observed; final generalization error showed low variability across seeds, consistent with the robustness analyses reported below.

Table A.5: Scattering Compositional Learner (SCL) architecture.

Stage	Module type	Dimension
Panel (CNN)	Shared convolutional block	3×80
Conv1	1-D convolution	80×64
Conv2	1-D convolution	80×32
Conv3	1-D convolution	80×5
MLP1	Fully connected layer	400
MLP2	Fully connected layer	400
Output	Classification head	35

Table A.6: Training-time computational cost on an NVIDIA H100.

Loss	Sec/epoch	ms/step	Samples/sec	Peak mem (GB)
SNR loss	520.30	802.3	637.6	2.10
VICReg	763.88	1178.8	434.3	2.68
CE	501.38	732.9	698.1	2.24
Prototypical	623.70	962.5	532.0	2.57

A.3 Model performance

A.3.1 Test performance

Test error was computed separately for each training rule using 1,000 independent evaluation samples and then averaged across rules within each split. Confidence intervals reported in Chapter 2 correspond to 95% confidence intervals across splits. All models performed significantly above chance level (error rate $= 1 - 1/35 \approx 0.971$), as assessed by one-sample t-tests across splits: ResNet ($t(14) = -312.77, p = 1.2 \times 10^{-28}$), WReN ($t(14) = -343.78, p = 3.4 \times 10^{-29}$), ViT ($t(14) = -388.11, p = 6.2 \times 10^{-30}$), and SCL ($t(14) = -542.53, p = 5.7 \times 10^{-32}$). Pairwise model comparisons were conducted using Wilcoxon signed-rank tests across splits, with Benjamini–Hochberg false discovery rate correction applied across the three comparisons. These analyses revealed a consistent performance ordering, with SCL achieving the lowest test error, followed by ViT, WReN, and ResNet (q-values: SCL vs. ViT = 4.6×10^{-5}; ViT vs. WReN = 1.7×10^{-3}; WReN vs. ResNet = 4.6×10^{-5}). Corresponding rank-biserial effect sizes were large to maximal (SCL vs. ViT: $r = 1.000$; ViT vs. WReN: $r = 0.817$; WReN vs. ResNet: $r = 1.000$).

A.3.2 Generalization performance

Generalization performance was evaluated on the *SimplifiedRPM* task. In each split, models were evaluated on five held-out rules, corresponding to 20 ordered held-out rule pairs. Generalization error was computed separately for each held-out rule pair using 500 independent *SimplifiedRPM* task trials and then averaged across rule pairs within each split. Confidence intervals reported in Chapter 2 correspond to 95% confidence intervals across splits.

All models performed significantly above chance level (error rate = 0.5), as assessed by one-sample t-tests across splits: ResNet ($t(14) = -8.93, p = 1.8 \times 10^{-7}$), WReN ($t(14) = -26.39, p =$

1.2×10^{-13}), ViT ($t(14) = -33.45, p = 4.6 \times 10^{-15}$), and SCL ($t(14) = -19.35, p = 8.3 \times 10^{-12}$). Pairwise model comparisons were conducted using Wilcoxon signed-rank tests across splits, with Benjamini–Hochberg false discovery rate correction applied. These analyses revealed significant differences between SCL and ViT ($q = 9.16 \times 10^{-5}$, rank-biserial $r = 0.983$) and between ViT and WReN ($q = 9.16 \times 10^{-5}$, $r = 1.000$), while no significant difference was observed between WReN and ResNet ($q = 0.34$, $r = 0.133$).

A.3.3 Relation-wise performance

Relation-wise differences in performance were evaluated across three abstract relation types (Progression, Arithmetic, and Logical) for four models (ResNet, WReN, ViT, and SCL) using 15 independent splits of training and held-out rules. Errors were averaged separately for each relation type within each split. Test performance is summarized in Fig. A.2. Statistical comparisons were performed using two-sided Mann–Whitney U tests on split-level relation-specific mean errors; effect sizes were quantified using Cliff's δ, and p-values were corrected for multiple comparisons using the Benjamini–Hochberg false discovery rate (FDR) procedure.

For test errors, ResNet, WReN, and ViT exhibited significant differences across all pairwise comparisons between relation types after correction ($q = 3.39 \times 10^{-6}$ for all comparisons, Cliff's $\delta = 1$). For SCL, Arithmetic relations yielded significantly higher test errors than Progression relations (Progression vs. Arithmetic: $q = 0.0433$, Cliff's $\delta = 0.413$), whereas Logical relations yielded significantly lower test errors than both Progression and Arithmetic relations (Arithmetic vs. Logical and Progression vs. Logical: $q = 4.41 \times 10^{-6}$, Cliff's $\delta = 1$).

For generalization errors, no pairwise differences were significant for ResNet after FDR correction (all $q > 0.05$). For WReN, Arithmetic relations produced significantly higher errors than both Progression ($q = 0.00825$, Cliff's $\delta = 0.672$) and Logical relations ($q = 0.00857$, Cliff's $\delta = 0.621$). In contrast, Progression and Logical relations did not differ significantly ($q = 0.59$,

Cliff's $\delta = 0.12$). For ViT, all pairwise comparisons were significant after correction (Progression vs. Arithmetic: $q = 8.3 \times 10^{-5}$, Cliff's $\delta = 0.94$; Arithmetic vs. Logical: $q = 3.7 \times 10^{-3}$, Cliff's $\delta = 0.65$; Progression vs. Logical: $q = 4.6 \times 10^{-4}$, Cliff's $\delta = 0.78$). For SCL, Progression relations differed significantly from both Arithmetic ($q = 0.013$, Cliff's $\delta = 0.59$) and Logical relations ($q = 2.1 \times 10^{-4}$, Cliff's $\delta = 0.86$). In contrast, Arithmetic and Logical relations did not differ significantly ($q = 0.55$, Cliff's $\delta = 0.14$).

A.3.4 Geometric decomposition of generalization performance

To quantify how training alters the geometric determinants of generalization, generalization SNR and its constituent components were compared between untrained and trained models for held-out rules. For each model, statistics were computed across 15 held-out rule splits, each containing five held-out rules (20 ordered rule pairs per split). Relational manifolds for each held-out rule were estimated using 1,000 test samples. SNR-related quantities were first averaged across rule pairs within each split and then averaged across splits. Comparisons between trained and untrained models were performed using the Wilcoxon signed-rank test, with p-values corrected across the four models for each metric using the Benjamini–Hochberg false discovery rate procedure.

Training significantly increased generalization SNR and centroid separation (*signal*) for all architectures (all $q = 6.1 \times 10^{-5}$, rank-biserial correlation $r = 1.00$). Training effects on dimensionality (D) differed across models: D increased significantly for ResNet and ViT (both $q = 8.1 \times 10^{-5}$, $r = 1.00$), showed no significant change for WReN ($q = 0.073$, $r = 0.53$), and decreased significantly for SCL ($q = 8.1 \times 10^{-5}$, $r = 1.00$). Signal–noise overlap decreased significantly with training for ResNet, WReN, and ViT (all $q = 8.1 \times 10^{-5}$, $r = 1.00$), and was also reduced for SCL ($q = 1.5 \times 10^{-3}$, $r = 0.87$), indicating reduced alignment between the centroid-separation direction and within-rule variability across architectures.

A.3.5 Robustness of layer-wise manifold dimensionality

To assess the robustness of the layer-wise manifold dimensionality dynamics reported in Chapter 2, we evaluated their reproducibility across both training splits and random initializations. For each model, the dimensionality $D(\ell)$ of held-out rule manifolds was computed at each layer, and the stability of the resulting layer-wise trajectories was quantified using pairwise Spearman rank correlations. Fig. A.3 illustrates seed-level robustness of layer-wise geometry, including generalization error, SNR, signal, and dimensionality.

The stability of layer-wise dimensionality trajectories across training splits was assessed first. SCL, ResNet, and WReN showed highly stable dimensionality dynamics (mean $\rho = 0.92$–0.99, nearly all significant after FDR correction), whereas ViT exhibited substantially lower stability (mean $\rho \approx 0.7$). Stability across random seeds with the rule split held fixed showed the same pattern: SCL, ResNet, and WReN exhibited highly consistent dimensionality trajectories across seeds (mean $\rho \approx 0.94$–0.95, 100% significant), whereas ViT again showed reduced consistency (mean $\rho \approx 0.74$, 8% significant), consistent with its weakly varying dimensionality across intermediate attention layers.

There is no uniform tendency for manifold dimensionality to decrease with depth across architectures. For instance, ViT exhibits increased dimensionality in its final layer, whereas SCL shows pronounced dimensionality compression in its final two MLP layers. To assess whether this compression reflects a robust architectural property rather than a split-specific artifact, targeted statistical tests were conducted on late-layer dimensionality changes in SCL. Specifically, the split-level change was defined as $\Delta D^{(s)} = D^{(s)}(\text{MLP2}) - D^{(s)}(\text{MLP1})$, where negative values indicate compression. Across all 15 splits, $\Delta D^{(s)}$ was consistently negative (confirmed by a two-sided Wilcoxon signed-rank test), demonstrating robust dimensionality compression at the MLP layer for held-out rule representations.

Finally, complementary intuition for these layer-wise geometric transformations is provided

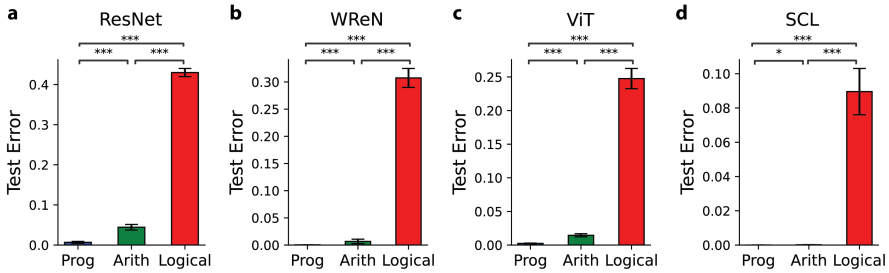


Figure A.2: Relation-wise test performance across models. (a–d) Test errors for four models (ResNet, WReN, ViT, and SCL) evaluated across three abstract relation types: Progression (blue), Arithmetic (green), and Logical (red). Results are computed over 15 splits of training and held-out rules, with test errors averaged separately for each relation type within each split. Bars show mean test error across splits, and error bars indicate 95% confidence intervals; the dashed line denotes chance-level error. Pairwise differences between relation types were assessed using two-sided Mann–Whitney U tests with false discovery rate correction and Cliff's δ effect sizes. ResNet, WReN, and ViT exhibit significant differences across all relation types. In contrast, SCL shows higher test errors for Arithmetic relations and lower test errors for Logical relations relative to the other relation types.

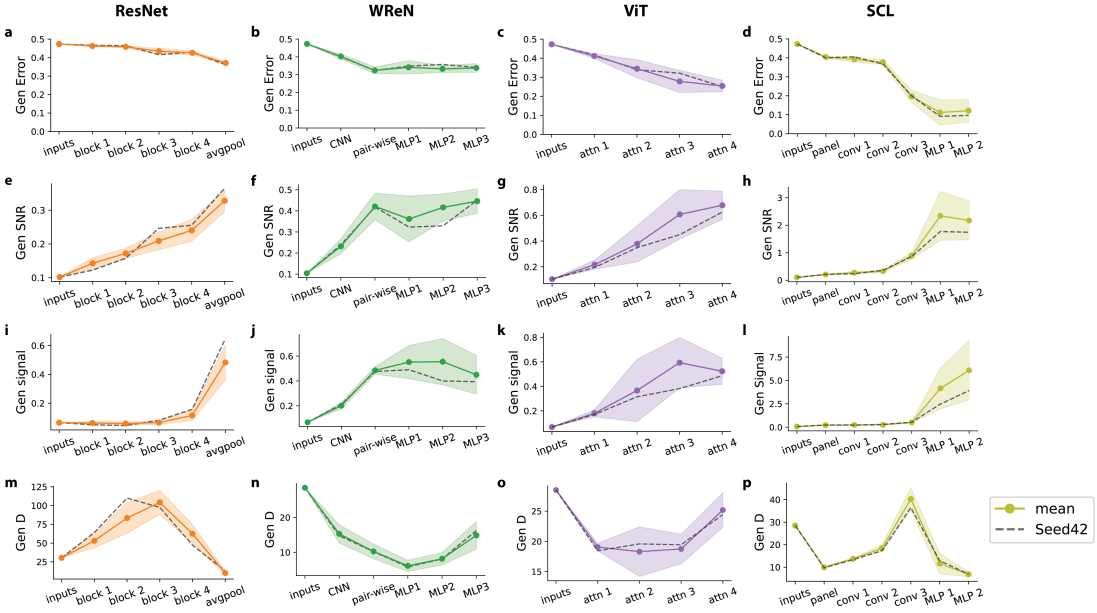


Figure A.3: Layer-wise generalization geometry is robust to random seed. Layer-wise generalization error (a–d), signal-to-noise ratio (SNR; e–h), signal (i–l), and dimensionality (D; m–p) across network layers for ResNet-50, WReN, ViT, and SCL. Results are shown for a fixed held-out rule split with training repeated using five random seeds. Solid lines indicate the mean across seeds, shaded regions denote 95% confidence intervals, and dashed black lines correspond to the seed used in Chapter 2. Qualitative layer-wise trends are consistent across seeds.

by two-dimensional PCA visualizations of held-out rule manifolds across layers for all models (Fig. A.4).

A.3.6 Perceptual and sampling factors in rule difficulty

To verify that the observed hierarchy of rule difficulty reflects intrinsic relational complexity rather than biases in attribute sampling, the analysis examined whether rule-dependent attribute variability could account for differences in generalization performance. Specifically, differences in the variability of visual attributes across rules could introduce unequal perceptual demands, confounding the interpretation of rule difficulty. Variability was measured in object attributes (object count and diversity of shapes, sizes, and shades) and low-level image statistics (foreground occupancy, edge density, luminance, and contrast), and it was assessed whether these measures predict rule-level generalization error or differ systematically across relation families.

For each of the 40 relational rules, within-rule variability was quantified using 1,000 held-out test rows. At the attribute level, we measured the total number of objects per row, as well as the number of unique shapes, sizes, and shades within each row, capturing variability in both object identity and numerosity. At the image level, we computed low-level statistics directly from the rendered panels, including foreground occupancy, edge density, mean luminance, and luminance contrast. For each rule and metric, we summarized variability by computing the variance of the corresponding measure across test samples, yielding a rule-wise estimate of attribute- and image-level variability. Both attribute-level and image-level variability exhibited substantial heterogeneity across rules. For example, at the attribute level, variance in the number of objects per row spanned more than a sixfold range across rules, while at the image level, the standard deviation of luminance contrast varied by more than a factor of five.

To test whether attribute-level variability explains rule difficulty, Spearman rank correlations were computed between rule-wise generalization error and each attribute-level variability. None of the

attributes reliably predicted rule difficulty: object count ($\rho = -0.22, p = 0.17$), number of unique sizes ($\rho = 0.30, p = 0.059, q_{\text{FDR}} = 0.094$), shapes ($\rho = -0.06, p = 0.70$), and shades ($\rho = 0.19$, $p = 0.24$) all showed non-significant correlations, with 95% confidence intervals including zero. These results suggest that attribute-level variability alone cannot account for the observed hierarchy of rule difficulty.

The next test asked whether attribute variability differs systematically across abstract relation families: Progression, Arithmetic, and Logical. For size, shape, and shade attributes, variability did not differ significantly between relation families, with all pairwise comparisons yielding large FDR-adjusted q-values ($0.48 \leq q \leq 1.0$) and small to modest Cliff's δ effect sizes ($-0.354 \leq \delta \leq 0.10$), indicating negligible differences.

In contrast, variability in the total number of objects differed markedly across relation families. Logical rules showed dramatically lower numerosity variability than Arithmetic rules ($q = 4.76 \times 10^{-5}$, Cliff's $\delta = 0.979$) and Progression rules ($q = 9.88 \times 10^{-6}$, Cliff's $\delta = 1.0$). This pattern runs counter to the model's generalization behavior: Logical rules are among the most difficult to generalize despite exhibiting the *lowest* numerosity variability. Moreover, numerosity variability does not differentiate Progression and Arithmetic rules, even though these two relation families reliably differ in generalization difficulty. Together, these results show that differences in numerosity variability cannot explain the observed hierarchy of rule difficulty.

Finally, we examined whether image-level variability drives rule difficulty. Across all measures, generalization error was negatively correlated with image-level variability, with Spearman correlations ranging from $\rho = -0.36$ to -0.60 across multiple image statistics (all $q_{\text{FDR}} < 0.05$). Thus, rules that are more difficult for the model to generalize are associated with *lower*, not higher, perceptual variability. These results indicate that rule difficulty is not explained by increased perceptual load; if anything, harder rules exhibit more constrained and less visually variable image statistics.

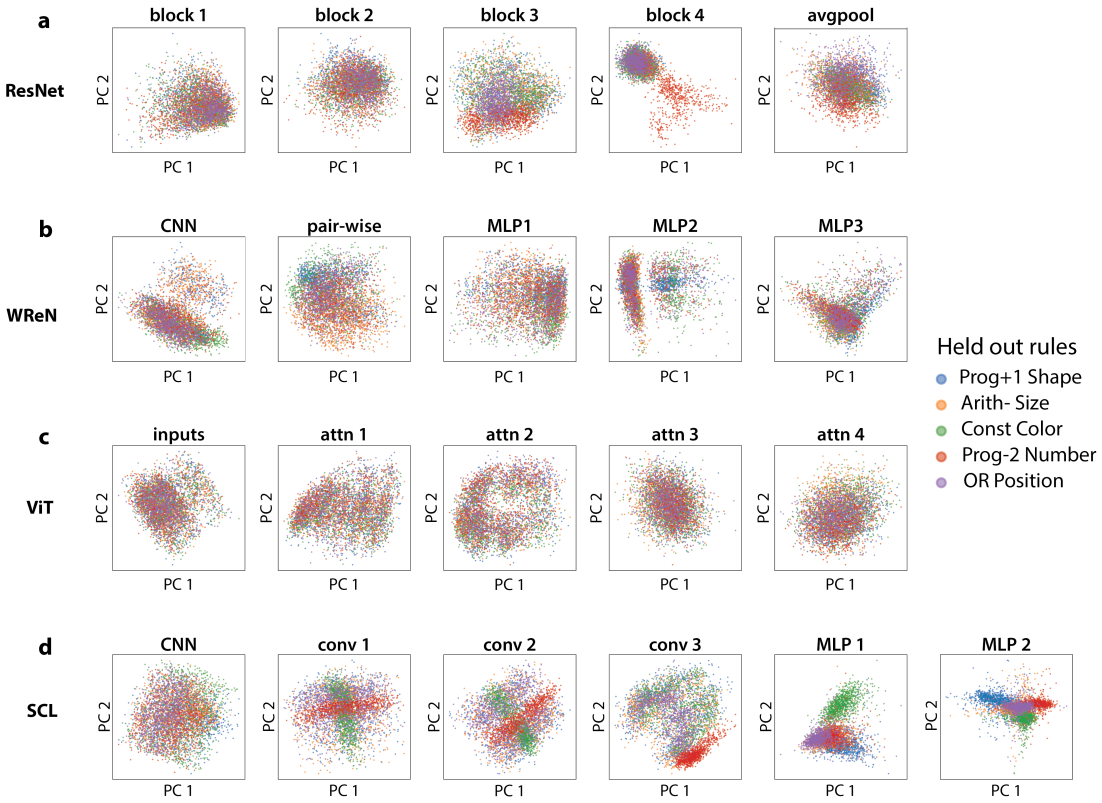


Figure A.4: Layer-wise PCA visualization of held-out rule manifolds. Two-dimensional PCA projections of layer-wise representations for (a) ResNet, (b) WReN, (c) Vision Transformer (ViT), and (d) SCL. Each point represents an input row representation, colored by one of five held-out relational rules. For each layer, PCA was fit on representations of the 35 training rules, and representations of held-out rules were projected into the same subspace. The visualization illustrates how held-out rule manifolds evolve across network layers.

A.4 Evaluation of Qwen-VL on *SimplifiedRPM*

A.4.1 Methods

A contemporary multimodal large language model, `Qwen/Qwen3-VL-8B-Instruct`, was evaluated on the SimplifiedRPM task under conditions closely matched to the human behavioral experiment and the neural network evaluation protocol. The model was used in inference-only mode, without fine-tuning or parameter updates. For each split, five relational rules were designated as held out, yielding 20 ordered rule pairs. For each ordered pair, 500 independent two-choice SimplifiedRPM trials were generated, resulting in 10,000 evaluation trials per split.

Trial construction followed the same procedure used for the trained neural architectures. For each ordered rule pair (a, b), a context row and a rule-consistent candidate row were sampled independently from held-out rule a, and a rule-inconsistent candidate row was sampled from held-out rule b. The two candidate rows were randomly assigned to the "Choice 1" and "Choice 2" positions. Each trial was rendered as a single composite image containing three labeled rows (Context Row, Choice 1, Choice 2), with panels rendered at 160×160 resolution using the same attribute sampling procedure described for the SimplifiedRPM dataset.

To match the human experimental setting as closely as possible, evaluation was conducted in a few-shot prompting framework using the same three example trials provided to human participants during the practice phase. These examples were presented before each test query in identical composite image format and included step-by-step explanations terminating in the correct choice label. The model was instructed to infer the abstract rule governing the Context Row and select the matching candidate. Responses were parsed by extracting the final occurrence of "Choice 1" or "Choice 2"; invalid or ambiguous responses were counted as incorrect. Chance performance in this setting is 0.5.

Generation was performed using greedy decoding (`do_sample = False`) with a maximum of 512

newly generated tokens. We loaded the model in `bfloat16` precision and evaluated it on a single NVIDIA H100 GPU. Composite trial images were provided to the model as base64-encoded PNG inputs within the multimodal chat template.

A.4.2 Prompt template

All evaluations used a fixed system instruction and user prompt. The system instruction was:

```
You are an expert at abstract visual pattern recognition.
Each image shows three labeled rows:

  - The top row is the "Context Row"
    (three panels that demonstrate an abstract rule).
  - The next two rows are "Choice 1" and "Choice 2",
    each consisting of three panels.

Identify the abstract rule from the Context Row.
Then determine which choice row follows the same rule.

Think step by step, then end your response with exactly
"Choice 1" or "Choice 2".
```

For each trial, the following user prompt was provided together with the composite image:

```
The image shows three rows:

  - Context Row (top): three panels that follow an abstract rule.
  - Choice 1 (middle): a candidate row of three panels.
  - Choice 2 (bottom): another candidate row of three panels.
```

```
Step 1: Identify the abstract rule demonstrated by the Context Row
        (e.g., a pattern in number, shape, size, color, or position
        across the three panels).

Step 2: Check whether Choice 1 or Choice 2 follows that same rule.

Step 3: State your final answer as "Choice 1" or "Choice 2".
```

A.4.3 Results

Under the held-out rule generalization protocol, Qwen3-VL-8B-Instruct achieved a mean generalization error of 0.446 ± 0.034 (95% CI across splits). Performance was above chance (error = 0.5) in the two-choice SimplifiedRPM setting. Still, it remained worse than that of the specifically trained architectures evaluated in Chapter 2, including the Scattering Compositional Learner (SCL) trained with cross-entropy (0.153 ± 0.019, 95% CI). Future work may investigate whether task-specific fine-tuning, alternative prompting strategies, or structured reasoning augmentations can further improve relational generalization in multimodal large language models.

A.5 Human experiment

To assess whether human difficulty patterns on the *SimplifiedRPM* task align with those exhibited by contemporary vision and reasoning models, we conducted a controlled behavioral experiment with human participants. We compared their performance with model generalization errors across a matched set of rule pairs.

A.5.1 Human experiment design

Participants were first introduced to the task through explicit instructions and three practice trials. During the practice phase, participants received feedback explaining the correct answer after each selection, ensuring they understood the task structure and response requirements before the main experiment. In the main experiment, participants completed trials without feedback (Fig. A.5).

Thirteen selected rule pairs were tested, spanning number-, shape-, and position-based relational rules. Each rule pair consisted of one correct rule governing both the sample row and the correct choice row, and one incorrect rule governing the incorrect choice row. Twenty-five Amazon Mechanical Turk participants ($n = 25$) were recruited, each completing a subset of 60 trials drawn from a total pool of 130 trials (13 rule pairs $\times$ 10 trials per rule pair). Trials were randomly sampled without replacement within each participant, and each trial was presented to multiple participants across the cohort. For each trial, the error was computed as the mean response error across 10 participants. Response time was measured as the duration from stimulus onset to the participant's selection.

Summary statistics for human performance—including mean error rate and response time for each rule pair—are reported in Table 2.2. Rule pairs are ordered by increasing human error rate.

A.5.2 Participants and exclusions

A total of 35 participants were initially recruited via Amazon Mechanical Turk. Participants were excluded based on three *a priori* criteria: (i) failure to complete the full experiment, (ii) failure on attention-check trials, and (iii) implausibly fast response times (RT $<$ 300 ms). To assess attention, the experiment included three attention-check trials, which were exact replicas of the instruction examples presented before the main experiment and for which the correct response was explicitly stated. These attention-check trials were interleaved with the main task, and participants had to

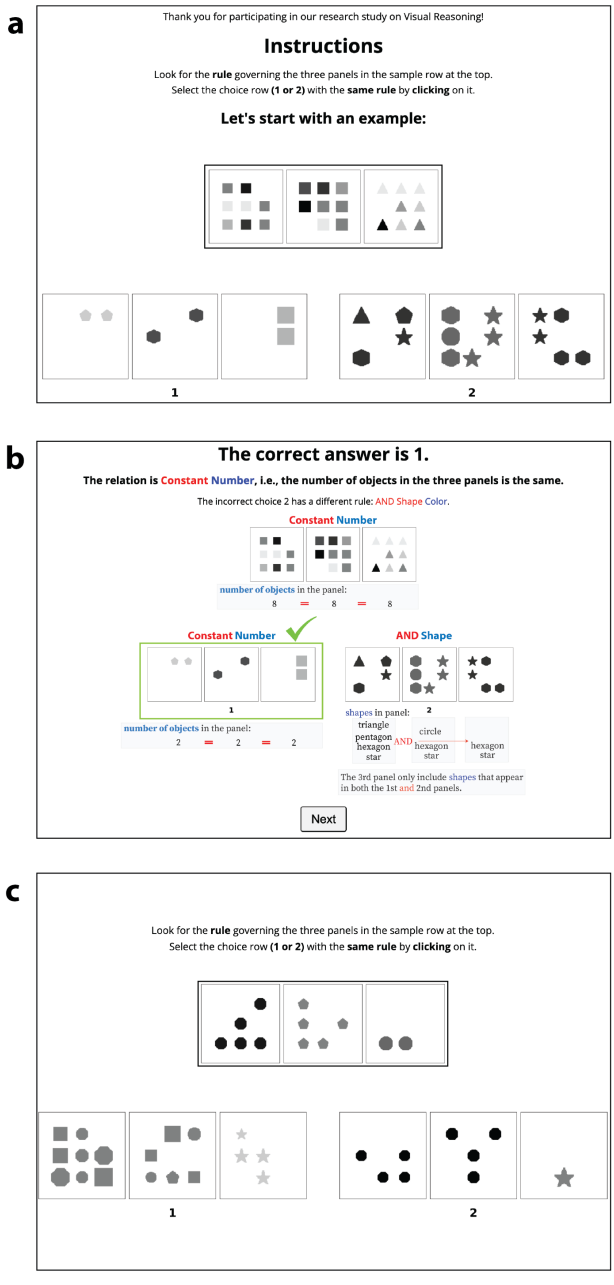


Figure A.5: Example trials in the human experiment. (a) Instructions and one of the three practice trials, in which participants are introduced to the task. (b) Explanations are provided to participants after they make a selection in a practice trial. (c) Example trial from the main experiment, where no feedback is given.

respond correctly to all three to be included. Three participants exited the experiment early and were excluded for incomplete participation; four were excluded for failing the attention checks; and an additional three were excluded for exhibiting implausibly fast response times. The final analyzed sample consisted of 25 participants. At the trial level, trials with excessively long response times were further excluded (RT $>$ 110 s, corresponding to the mean plus three standard deviations), which likely reflect lapses of attention or task disengagement rather than task-related computation.

A.6 Human–model comparison

Human error rates were compared with the generalization errors of the three remaining models, ResNet-50, WReN, and ViT, on the same 13 rule pairs (Fig. A.6a–c). Each point represents the average model error (x-axis) and human error (y-axis) across the 10 trials associated with a given rule pair, with error bars denoting 95% confidence intervals. For ResNet-50, WReN, and ViT, there was no significant positive correlation between model and human error patterns (Spearman's rank correlation: ResNet-50, $\rho = 0.34, p = 0.24$; WReN, $\rho = -0.67, p = 0.01$; ViT, $\rho = 0.27$, $p = 0.37$).

To further examine whether model difficulty aligns with human cognitive effort, the relationship between model generalization error and human response time was analyzed for each rule pair (Fig. A.6d–f). No significant correlation was found between model error and human response time for any of the models tested (Spearman's rank correlation: ResNet-50, $\rho = 0.12, p = 0.68$; WReN, $\rho = -0.50, p = 0.08$; ViT, $\rho = -0.14, p = 0.65$).

Together, these results indicate that, despite comparable overall performance levels, current models and human participants exhibit markedly different difficulty profiles across relational rule combinations. For these three architectures, therefore, model generalization error does not capture the structure of human reasoning difficulty on the *SimplifiedRPM* task; SCL is the exception (Sec-

tion 2.4).

A.6.1 Response time normalization

To control for individual differences in overall response speed, RTs were log-transformed and z-scored within each participant before averaging by rule pair. The resulting normalized RTs remained significantly correlated with the SCL model's generalization error (Spearman $\rho = 0.62, p = 0.024$), indicating that the RT–model relationship reflects intrinsic rule difficulty rather than participant-specific speed differences.

A.6.2 Perceptual difficulty control

To test whether variability in human performance across rule pairs could be explained by low-level perceptual difficulty rather than relational reasoning demands, perceptual difficulty was quantified using four complementary metrics: total number of objects (the count of all objects across panels), total object area (the sum of object pixel areas), number of unique shapes, and number of unique sizes. Together, these metrics capture complementary aspects of perceptual load. For each rule pair, all metrics were averaged across the three rows and the ten questions associated with that rule pair to yield a single summary value.

Whether human error rates across rule pairs were associated with perceptual difficulty was examined by computing Spearman rank correlations between error rates and each perceptual metric. No significant relationships were observed (total number of objects: $\rho = -0.17, p = 0.58$; total object area: $\rho = -0.13, p = 0.68$; number of unique shapes: $\rho = 0.02, p = 0.52$; number of unique sizes: $\rho = -0.15, p = 0.60$). Similarly, no significant correlations were observed between human response times (RTs) and perceptual metrics (total number of objects: $\rho = 0.33, p = 0.27$; total object area: $\rho = 0.39, p = 0.19$; number of unique shapes: $\rho = 0.39, p = 0.19$; number of unique

sizes: $\rho = 0.51, p = 0.058$). Nonparametric bootstrap resampling of rule pairs (10,000 iterations) yielded 95% confidence intervals that included zero for all correlations, indicating that neither human error rates nor response times are robustly explained by perceptual difficulty as quantified by these measures.

A.6.3 Human–model correspondence controlling for perception

To determine whether the correspondence between human behavior and model generalization performance could be explained by low-level perceptual difficulty, partial correlation analyses were performed controlling for all four perceptual metrics. Specifically, we examined the association between human behavior and the SCL model's generalization error across rule pairs after regressing out the effects of total number of objects, total object area, number of unique shapes, and number of unique sizes.

After controlling for perceptual difficulty, the correspondence between human behavior and model generalization performance remained strong across rule pairs. Human error rates showed a robust positive association with model generalization error after regressing out all four perceptual metrics (partial Spearman $\rho = 0.74, p = 0.004$). Human response times exhibited an even stronger relationship under the same controls (partial Spearman $\rho = 0.93, p = 3 \times 10^{-6}$). These results indicate that the alignment between human behavior and model performance does not reflect low-level perceptual discriminability, but instead reflects shared sensitivity to higher-level task demands.

A.6.4 Robustness of human–model correspondence in difficulty ordering

To assess the robustness of the human–model correspondence in difficulty ordering, a series of leave-one-out analyses was performed at both the rule-pair and relation-family levels. In a leave-one-rule-pair-out analysis, the correlation was recomputed between human behavior and model

generalization error after excluding each of the 13 rule pairs in turn. Across all iterations, the correlations remained consistently high ($\rho = 0.88$–0.92, all $p < 4 \times 10^{-5}$), indicating that the observed correspondence is not driven by any individual rule pair. Rule pairs were further grouped into three abstract relation families—Progression, Arithmetic, and Logical—and the analysis was repeated while excluding all rule pairs from one family at a time. The human–model correlation remained significant in all cases (excluding Arithmetic: $\rho = 0.875, p = 4.25 \times 10^{-4}$; excluding Logical: $\rho = 0.786, p = 0.036$; excluding Progression: $\rho = 0.719, p = 0.045$), demonstrating that the effect persists across relation families and is not specific to any particular class of relational rules.

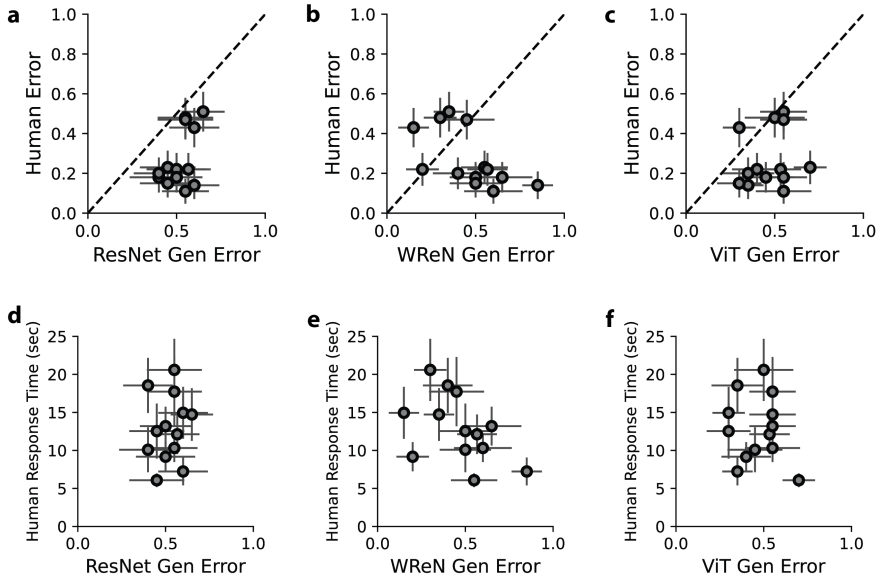


Figure A.6: Comparison of model generalization errors with human performance on the *SimplifiedRPM* task. (a–c) Comparison of the generalization errors of ResNet-50, WReN, and ViT models with human performance on the *SimplifiedRPM* task across 13 selected rule pairs. Each dot represents the average model error (x-axis) and human error (y-axis) over 10 questions for a given rule pair. Error bars indicate 95% confidence intervals. The dashed line represents the line of identity. No significant positive correlation is observed between model and human errors (Spearman's rank correlation: ResNet-50, $\rho = 0.34, p = 0.24$; WReN, $\rho = -0.67, p = 0.01$; ViT, $\rho = 0.27, p = 0.37$). (d–f) Relationship between model generalization error (x-axis) and human response time (y-axis) for the 13 selected rule pairs in ResNet-50, WReN, and ViT models. Each dot represents the average model error and corresponding human response time for a given rule pair, with error bars indicating the 95% CI. No significant correlation is found between model error and human response time (Spearman's rank correlation: ResNet-50, $\rho = 0.12, p = 0.68$; WReN, $\rho = -0.50, p = 0.08$; ViT, $\rho = -0.14, p = 0.65$).

A.7 Extension towards canonical RPM

The canonical Raven's Progressive Matrices (RPM) task consists of a 3×3 matrix completion problem in which the missing panel is selected from eight candidates based on an underlying relational structure (Fig. A.7a). While *SimplifiedRPM* isolates single-row relational abstraction to enable precise geometric analysis, the canonical RPM format introduces additional structural demands, including inference across multiple rows and multi-choice decision-making. The proposed geometric framework is not specific to *SimplifiedRPM* and extends to more complex inference procedures.

A.7.1 Task setup

Relational generalization is evaluated in a task format that more closely resembles the canonical RPM setting (Fig. A.7a). Each trial consists of a 3×3 matrix in which the first two rows are generated according to a common relational rule (rule a). The model is then presented with eight candidate rows. One candidate is generated from the same rule a, while the remaining seven candidates are sampled from a different relational rule (rule b). The task is to identify the candidate row that belongs to the same relational rule as the first two rows. This formulation is a first-step extension beyond *SimplifiedRPM* that probes whether the learned relational representations support comparison and generalization across multiple rows. More complex extensions that more closely mirror the full canonical RPM structure, including partial-row completion and hierarchical rule composition, are left for future work.

Trained SCL models are evaluated on this more complex task in a zero-shot manner, without any additional training. Each row is encoded into a relational representation using the model encoder. The representations of the two reference rows are averaged to form a prototype representation for rule a. Each of the eight candidate rows is then compared to this prototype, and the candidate with the minimal distance is selected as the model's prediction (schematized in Fig. A.7b). Performance is

quantified as classification accuracy over eight choices, with chance level at 12.5%. The eight-choice RPM completion error is defined as the fraction of incorrect selections across evaluation trials.

A.7.2 Extension of the geometric framework

Extending the geometric SNR framework to this more structured inference setting requires two modifications to the decision process: multi-choice selection and prototype averaging across multiple reference rows. Multi-choice decisions are handled by expressing the trial error as a function of the Gaussian tail probability associated with the margin distribution. With eight total choices, an error occurs if the margin is negative for at least one of the $n_{\text{choice}} = 7$ candidate rows independently sampled from rule b. Assuming independence, the resulting eight-choice RPM completion error is

$$\varepsilon_{a,b} = 1 - \big(1 - H(\text{SNR}_{a,b})\big)^7. \tag{A.3}$$

The use of two reference rows corresponds to the m-shot prototype learning setting in the geometric framework. The resulting signal-to-noise ratio for the averaged prototype has been derived in prior work[15]:

$$\text{SNR}_m = \frac{1}{2} \frac{\|\Delta \mathbf{x}_0\|^2 + \left(\frac{R_b^2}{R_a^2} - 1\right)/m}{\sqrt{D_a^{-1}/m + \|\Delta \mathbf{x}_0 \cdot \mathbf{U}_b\|^2/m + \|\Delta \mathbf{x}_0 \cdot \mathbf{U}_a\|^2}}, \tag{A.4}$$

which is applied here directly with $m = 2$.

In summary, the task is predicted by the geometric SNR framework using the m-shot prototype SNR with $m = 2$ (Eq. A.4) combined with an eight-choice decision rule ($n_{\text{choice}} = 7$), yielding a trial error $\varepsilon_a = 1 - (1 - H(\text{SNR}_{m=2}))^7$.

A.7.3 Results

For the SCL model trained with cross-entropy loss, zero-shot generalization was evaluated on the more complex 3×3 RPM task across five held-out rule splits. The mean eight-choice RPM completion error was 0.1690 ± 0.0181 (mean $\pm$ s.e.m.), well below the chance error rate of 0.875. The geometric signal-to-noise ratio for two-shot prototype inference, $\mathrm{SNR}_{m=2}$, was additionally quantified and had a mean value of 3.0236 ± 0.2666 in the 3×3 task. The observed eight-choice RPM completion error is well described by the measured $\mathrm{SNR}_{m=2}$ (Fig. A.7c), supporting the applicability of the geometric framework to the more complicated RPM inference setting. This analysis demonstrates the applicability of the proposed geometric framework to a more complex task setup towards the canonical RPM format. The current task setting does not constitute full RPM reasoning and does not address interactions or composition among multiple relational rules, which is left for future work.

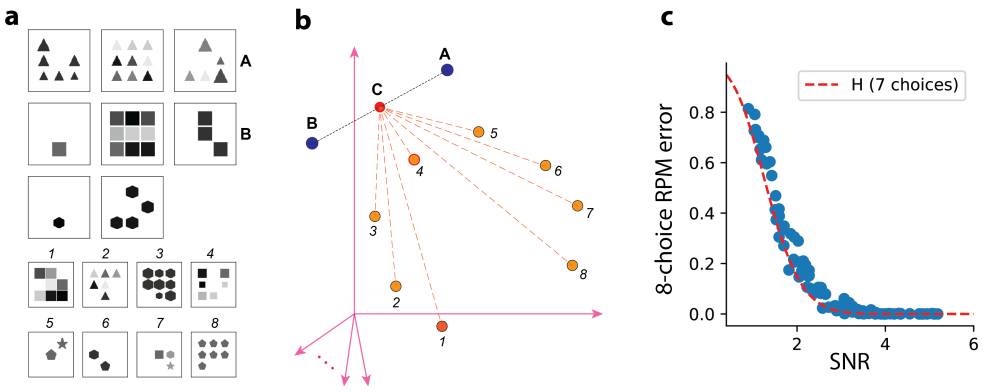


Figure A.7: Zero-shot extension to 3×3 Raven-style matrix completion and geometry–performance relationship. (a) Example 3×3 matrix completion task constructed from a single underlying relational rule. The bottom-right panel is removed, and eight candidate rows are provided; only one correctly completes the third row. (b) Schematic illustration of zero-shot inference in relational representation space. A prototype representation is formed by averaging the representations of the two completed rows (A and B). Each candidate completion (1–8) is scored by its Euclidean distance to the prototype, and the closest candidate is selected. (c) Eight-choice RPM completion error as a function of geometric signal-to-noise ratio (SNR) in the 3×3 task. Each point corresponds to a held-out rule pair evaluated zero-shot. The dashed curve shows the theoretical prediction for an eight-choice decision, $1 - (1 - H)^7$ with seven independent distractors. The SNR–error relationship is qualitatively consistent with the geometric theory, with a shifted mapping relative to *SimplifiedRPM* due to the increased choice set.

A.8 Combinatorial rule composition

A.8.1 Task setup

SimplifiedRPM enforces a one-rule-per-row constraint, ensuring clean factorization of rule manifolds. To evaluate whether the geometric framework remains valid when relational structure becomes combinatorially entangled through attribute overlap, we constructed a controlled compositional extension using held-out rules.

The same pretrained SCL models trained with cross-entropy loss in the main experiments were used. For each split, the model was trained on 35 single relational rules and evaluated on 5 held-out rules. From the five held-out rules $\{r_1, r_2, r_3, r_4, r_5\}$, three composite rules were generated per split by pairing two distinct held-out rules applied to different attributes, yielding six ordered composite rule pairs per split and 30 ordered composite rule pairs in total across five splits. Composite rules shared one relational component while differing in another (e.g., Const Shape + XOR Color vs. Const Shape + AND Color), thereby introducing controlled combinatorial attribute overlap and partial manifold entanglement. An example showing Const Shape + XOR Color is shown in Fig. A.8a.

A.8.2 Results

Across the 30 ordered composite rule pairs evaluated using five independently pretrained SCL models (five held-out splits), composite rule discrimination yielded a mean SNR of 1.073 ± 0.204 and a mean generalization error of 0.237 ± 0.031 (mean $\pm$ s.e.m.). Generalization error across composite rule pairs remained well described by the geometric prediction $\varepsilon \approx H(\text{SNR})$ (Fig. A.8b). These results demonstrate that the geometric framework continues to accurately account for generalization even when rule manifolds become partially entangled through combinatorial attribute overlap.

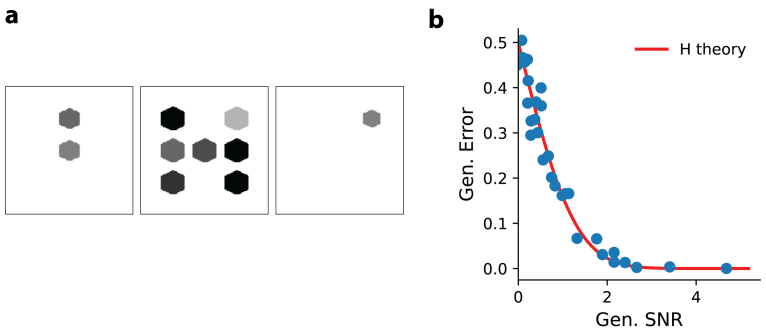


Figure A.8: Geometric prediction under combinatorial rule composition. (a) Example of a compositional relational row constructed by simultaneously applying two held-out rules (Constant Shape and XOR Color). (b) Generalization error as a function of geometric signal-to-noise ratio (SNR) for composite rule discrimination evaluated zero-shot. Results are shown for five pretrained SCL models (five held-out splits); in each split, three composite rules yield six ordered rule pairs. Each point represents one ordered composite pair. The red curve shows the theoretical prediction $\varepsilon \approx H(\mathrm{SNR})$.

A.9 Rule-dependent geometry in SCL

A potential explanation for SCL's correspondence with human difficulty ordering is that it simply enforces more uniform geometry across rules, rather than preserving meaningful rule-specific structure. To test this possibility, the coefficient of variation (CV) of rule-level generalization SNR was quantified across the five held-out rules within each training split, yielding a scale-normalized measure of across-rule heterogeneity.

Contrary to a uniform-geometry account, SCL does not exhibit reduced rule-level heterogeneity. Instead, SCL shows significantly greater across-rule variability in generalization SNR than ViT (paired Wilcoxon signed-rank test across splits, FDR-corrected $p = 0.0018$), with mean CV values of 0.612 for SCL versus 0.431 for ViT. In contrast, no significant differences in across-rule heterogeneity are observed between SCL and either ResNet or WReN (all $p > 0.6$), indicating that SCL is not atypical in its degree of rule-level variability.

Notably, ViT—despite exhibiting the most uniform rule geometry—fails to reproduce the human ordering of relational difficulty, whereas SCL does. This dissociation indicates that uniform representational geometry is not sufficient to account for the observed human–model correspondence in difficulty ordering and suggests that the preserved rule-to-rule heterogeneity in SCL reflects functionally meaningful structure rather than noise.

A.9.1 SCL selectively amplifies rule-dependent generalization signal

While the coefficient-of-variation analysis rules out a uniform-geometry explanation, it does not identify which geometric component underlies SCL's structured rule-level heterogeneity. We therefore repeated the analysis using the rule-level generalization signal, defined as the normalized separation between rule-manifold centroids.

For each training split, the coefficient of variation of the rule-level signal was quantified across

the five held-out rules, and models were compared using paired Wilcoxon signed-rank tests. Strikingly, SCL exhibited substantially greater signal heterogeneity than all other architectures (ResNet, WReN, and ViT; all $p < 3 \times 10^{-4}$). These results indicate that the observed rule-level heterogeneity in SCL is associated with differences in signal across rules.

A.9.2 SCL does not compress dimensionality uniformly across rules

To further assess whether SCL's correspondence with human difficulty ordering could be attributed to a uniform reduction in representational dimensionality, the coefficient-of-variation analysis was repeated using the dimensionality (D). For each training split, the coefficient of variation of D was computed across the five held-out rules in each split, and models were compared using paired Wilcoxon signed-rank tests.

SCL did not differ significantly from ResNet in across-rule dimensionality heterogeneity ($p = 0.93$). In contrast, SCL exhibited greater dimensional heterogeneity than both WReN ($p = 0.0013$) and ViT ($p = 0.0040$). These results indicate that, although SCL compresses representations at the level of individual layers, it preserves rule-dependent differences in dimensionality rather than enforcing uniformly low-dimensional representations.

A.9.3 Layer-wise emergence of human–model difficulty correspondence

To localize where human–model correspondence emerges within SCL, a layer-wise analysis was conducted of rule-level generalization error for the three rule families (Progression, Arithmetic, Logical). At the input stage, generalization error is high and similar across rules, with no clear ordering. After panel-wise encoding, Logical shows the lowest error, while Progression and Arithmetic show comparable, higher error.

Across the first two convolutional layers, Progression error decreases to a level similar to Logical,

whereas Arithmetic remains the most difficult rule type. In the final convolutional layer, generalization error decreases for all rules, but Logical increases relative to Progression and becomes comparable to Arithmetic.

A stable ordering first emerges at the first MLP layer. At this stage, Progression shows substantially lower generalization error than both Arithmetic and Logical, with Arithmetic intermediate and Logical highest. This ordering (Progression $<$ Arithmetic $<$ Logical) is preserved in the second MLP layer. Thus, correspondence with the human difficulty ordering emerges only after the transition from convolutional to MLP processing.

A.10 Structured compression

A.10.1 Statistical analysis

For each architecture, the relationships were quantified between the generalization signal and dimensionality (D) and between the generalization signal and alignment to the training-rule subspace, across 15 held-out rule splits. For each split, Spearman's rank correlation coefficient was computed across training epochs. To assess statistical significance at the model level, split-wise correlation coefficients were combined using Stouffer's method.

For the relationship between generalization signal and dimensionality, a negative correlation across splits was observed in all four architectures (mean $\pm$ s.e.m.: ResNet, $\rho = -0.12 \pm 0.12$; WReN, $\rho = -0.41 \pm 0.11$; ViT, $\rho = -0.37 \pm 0.19$; SCL, $\rho = -0.77 \pm 0.05$). This effect was statistically significant in all models.

The relationship between the generalization signal and alignment to the training-rule subspace was examined next. Subspace alignment was quantified as the fraction of variance in held-out rule representations explained by the top 35 PCs of the training-rule representation space. Principal component analysis of the training-rule representations revealed a pronounced drop in explained

variance at the 35th principal component across architectures (Fig. A.9), corresponding to the number of training rules and indicating that training-rule representations are confined to a low-dimensional subspace.

Across held-out splits, the generalization signal was positively correlated with subspace alignment in all architectures (mean ± s.e.m.: ResNet, $\rho = 0.01 \pm 0.13$; WReN, $\rho = 0.29 \pm 0.11$; ViT, $\rho = 0.38 \pm 0.19$; SCL, $\rho = 0.90 \pm 0.04$). This effect was statistically significant for ResNet, ViT, and SCL, but not for WReN.

A.10.2 Polarization regularization

To provide more direct evidence for the structured compression hypothesis, we introduced an explicit geometric regularization during training that biases representations toward structured compression. Specifically, we added a polarization regularizer to the cross-entropy (CE) loss that encourages variance to concentrate along fewer directions in feature space. Let $\mathbf{x}_i \in \mathbb{R}^D$ denote the relational representation of row i within a mini-batch of size B. Features were mean-centered within each batch, and the empirical covariance matrix was computed as

$$\Sigma = \frac{1}{B-1} \sum_{i=1}^{B} (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^\top. \tag{A.5}$$

A scale-invariant polarization objective was defined

$$\mathcal{L}_{\text{pol}} = -\frac{\|\Sigma\|_F^2}{(\text{Tr}(\Sigma))^2 + \varepsilon}, \tag{A.6}$$

where $\| \cdot \|_F$ denotes the Frobenius norm, $\text{Tr}(\cdot)$ the trace, and $\varepsilon = 10^{-8}$ ensures numerical stability. This objective increases as variance becomes concentrated along fewer principal directions and is invariant to uniform rescaling of the representation.

The total training objective was

$$\mathcal{L} = \mathcal{L}_{\text{CE}} + \gamma \mathcal{L}_{\text{pol}}, \tag{A.7}$$

with $\gamma = 0.01$. Minimizing this objective encourages the learned representations to concentrate their variance along a small number of principal directions, while the CE loss preserves performance on the task.

Using the same training protocol as the CE baseline, SCL models were trained across five held-out generalization splits. Polarization regularization reduced dimensionality while increasing centroid separation (signal), though generalization error was slightly higher than the CE baseline (0.122 vs. 0.099; Table A.7). These results show that directly biasing representations toward low-dimensional structure selectively modulates representation geometry in the predicted direction, supporting the structured compression hypothesis.

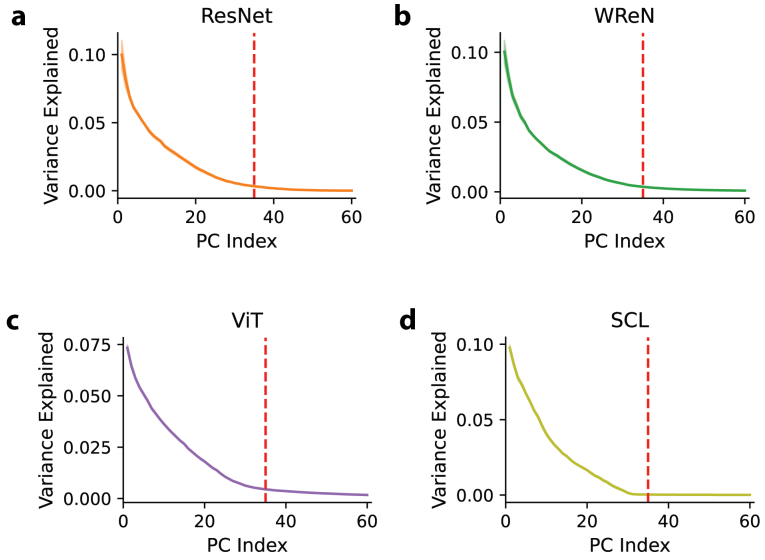


Figure A.9: Fraction of variance explained by the principal components (PCs) of the training-rule representations. (a–d) The relational representations of the 1,000 test rows for each of the 35 training rules are aggregated, and principal component analysis (PCA) is performed on them. For each of the four neural networks under study, the panel shows the fraction of variance explained by each principal component (PC), displaying only the top 60 PCs. The red line indicates the 35th PC, corresponding to the number of training rules. A noticeable drop in explained variance at this point suggests that the representations of the 35 training rules are consistently confined to a low-dimensional subspace. The lines represent the average explained variance across 15 random splits of held-out rules, with shading indicating the standard error of the mean (s.e.m.).

Table A.7: Polarization regularization ablation. Values are mean $\pm$ s.e.m. across five held-out splits for generalization error, estimated generalization SNR, centroid separation (signal), and dimensionality D.

Training Objective	Gen. Error	Gen. SNR	Gen. Signal	Gen. D
CE baseline	0.099 ± 0.015	2.01 ± 0.13	5.06 ± 0.59	9.12 ± 0.85
CE + polarization	0.122 ± 0.018	2.19 ± 0.14	6.69 ± 0.72	8.49 ± 0.92

A.11 Training with SNR loss

A.11.1 Hyperparameter selection

A grid search was performed over batch compositions $\{32 \times 16,\ \ 16 \times 32,\ \ 8 \times 64\}$ (rules $\times$ samples per rule), learning rates $\{0.1, 0.01, 0.001\}$, and temperature values $\alpha \in \{0.1, 0.5, 1.0, 2.0\}$. Model performance was assessed using generalization error on three held-out rule splits. Table A.8 reports the dependence of generalization error and geometric quantities on α. Based on this search, we selected a batch composition of 32×16, a learning rate of 0.01, and $\alpha = 0.5$.

A.11.2 Optimization stability

SNR training was evaluated across five random seeds using identical sampling and optimization settings. All runs completed without numerical instabilities, NaNs, or divergence. Final generalization error across seeds was 0.129 ± 0.008 (mean $\pm$ s.d.).

To assess representational stability, effective manifold dimensionality and feature variance were monitored during training. Dimensionality remained strictly positive and did not collapse across seeds, confirming that SNR optimization does not induce degenerate representational collapse. This behavior is consistent with the objective structure: collapsed representations yield vanishing centroid separation and reduced dimensionality, resulting in low SNR and therefore high loss.

A.11.3 Prototypical loss

The Prototypical loss[73] is a distance-based objective commonly used in few-shot learning. It constructs a prototype representation for each relational rule, which serves as a representative embedding for that rule. As in SNR loss training, each mini-batch consists of m relational rules with P rows per rule. For each rule r, the P rows are randomly partitioned into two disjoint subsets: a sup-

port set S_r and a query set Q_r.

The prototype z_r for rule r is computed as the mean representation of the support set,

$$z_r = \frac{1}{|S_r|} \sum_{x \in S_r} x, \tag{A.8}$$

where x denotes the relational representation of an input row.

The Prototypical loss encourages query samples to be close to their corresponding rule prototype while being distant from prototypes of other rules in the batch. It is defined as

$$\mathcal{L}_{\text{Proto}} = \sum_{r} \sum_{x \in Q_r} -\log \frac{\exp\left(-\|x - z_r\|^2\right)}{\sum_{c \neq r} \exp(-\|x - z_c\|^2)}. \tag{A.9}$$

To optimize the Prototypical loss, a grid search was performed over batch compositions $\{32 \times 16,\ 16 \times 32,\ 8 \times 64\}$ (rules $\times$ samples per rule), learning rates $\{0.1, 0.01, 0.001\}$, and support set sizes $\{1, 5\}$. We selected a batch composition of 32×16, a learning rate of 0.01, and a support set size of 5.

A.11.4 VICReg

VICReg[74,75] encourages learned representations to maintain high variance across feature dimensions while reducing redundancy by penalizing cross-dimensional covariance, without explicitly optimizing pairwise rule separability.

Each mini-batch consists of m relational rules, each with P rows. Let $\mathbf{x}_i \in \mathbb{R}^D$ denote the representation of the i-th input row in a batch of size $B = mP$. Representations are first mean-centered within the batch,

$$\tilde{\mathbf{x}}_i = \mathbf{x}_i - \frac{1}{B} \sum_{j=1}^{B} x_j. \tag{A.10}$$

The variance regularization term encourages each feature dimension to maintain sufficient vari-

ability,

$$\ell_{\text{var}} = \frac{1}{D}\sum_{k=1}^{D} \max\Big(0,\ 1 - \sqrt{\text{Var}(\tilde{x}_{\cdot k}) + \varepsilon}\Big)\,, \tag{A.11}$$

where $\tilde{x}_{\cdot k}$ denotes the k-th feature dimension across the batch and ε is a small constant for numerical stability. The covariance regularization term penalizes correlations between different feature dimensions,

$$\ell_{\text{cov}} = \frac{1}{D(D-1)}\sum_{i\neq j} C_{ij}^2, \tag{A.12}$$

where C is the empirical covariance matrix of the centered representations,

$$C = \frac{1}{B-1}\tilde{X}^\top \tilde{X}, \tag{A.13}$$

and $\tilde{X} \in \mathbb{R}^{B\times D}$ stacks the centred representations.

The full training objective combines cross-entropy loss with variance-covariance regularization,

$$\mathcal{L}_{\text{VICReg}} = \mathcal{L}_{\text{CE}} + \alpha\,\ell_{\text{var}} + \beta\,\ell_{\text{cov}}. \tag{A.14}$$

A grid search was performed over batch compositions $\{32\times 16,\ 16\times 32,\ 8\times 64\}$ (rules $\times$ samples per rule), learning rates $\{0.1, 0.01, 0.001\}$, and regularization strengths $\alpha \in \{0.1, 1.0\}$ and $\beta \in \{0.01, 0.1\}$. The selected configuration was a batch composition of 32×16, a learning rate of 0.01, $\alpha = 1.0$, and $\beta = 0.01$.

A.11.5 Pairwise comparison statistics

Pairwise comparisons across loss functions revealed systematic differences in generalization error, signal, and dimensionality (D). Table A.9 shows summary statistics for generalization error, signal, dimensionality, and SNR across all objectives. SNR loss exhibited significantly lower generalization

error than VICReg ($q = 1.11 \times 10^{-4}, r = 0.261$), CE ($q = 1.65 \times 10^{-8}, r = 0.381$), and Prototypical loss ($q = 4.56 \times 10^{-36}, r = 0.845$). Additional comparisons showed higher error for VICReg relative to CE ($q = 0.0163, r = 0.163$) and for Prototypical loss relative to all other objectives ($q < 10^{-13}$).

For signal strength, SNR loss showed lower signal than CE ($q = 7.26 \times 10^{-4}, r = 0.228$) and VICReg ($q = 0.0261, r = 0.148$), but substantially higher signal than Prototypical loss ($q = 9.51 \times 10^{-45}, r = 0.940$). The prototypical loss differed significantly from all other objectives ($q < 10^{-41}$). In contrast, SNR loss achieved significantly higher dimensionality than VICReg ($q = 2.92 \times 10^{-4}, r = 0.481$) and CE ($q = 3.99 \times 10^{-11}, r = 0.891$), while remaining lower than Prototypical loss ($q = 1.11 \times 10^{-10}, r = 0.861$). Dimensionality differences between all remaining pairs were also significant ($q < 10^{-10}$). Mean generalization error (mean $\pm$ half-width of confidence interval) was lowest for SNR loss (0.096 ± 0.015), followed by VICReg (0.126 ± 0.017), CE (0.153 ± 0.019), and Prototypical loss (0.228 ± 0.015).

A.11.6 SNR loss ablation

To isolate the contributions of individual geometric components in the SNR loss, several ablated variants were evaluated in which specific terms of the SNR definition were selectively removed.

Signal-only ablation. Centroid separation alone is optimized, using the loss

$$\mathcal{L}_{\text{signal}} = \sum_{(a,b)} \exp\left(-\alpha \, \|\Delta \mathbf{x}_0\|^2\right), \tag{A.15}$$

where $\Delta \mathbf{x}_0 = (\mathbf{x}_a^0 - \mathbf{x}_b^0)/R_a$ denotes the normalized centroid difference between the two rule manifolds. Here $\alpha = 0.5$. This ablation tests whether directly maximizing centroid separation, without explicitly controlling manifold geometry, is sufficient to yield robust relational generalization.

No-signal ablation. To assess the contribution of this signal term, it was removed from the SNR:

$$\mathrm{SNR}_{a,b}^{\text{no-signal}} = \frac{\varepsilon + \left(\frac{R_b^2}{R_a^2} - 1\right)}{\sqrt{D_a^{-1} + \|\Delta \mathbf{x}_0 \cdot \mathbf{U}_a\|^2 + \|\Delta \mathbf{x}_0 \cdot \mathbf{U}_b\|^2}}. \tag{A.16}$$

Here $\varepsilon = 0.1$. The corresponding loss is given by

$$\mathcal{L}_{\text{no-signal}} = \sum_{(a,b)} \exp\left(-\alpha\, \mathrm{SNR}_{a,b}^{\text{no-signal}}\right). \tag{A.17}$$

No-bias ablation. To evaluate the contribution of the bias term, it was removed from the SNR:

$$\mathrm{SNR}_{a,b}^{\text{no-bias}} = \frac{\|\Delta \mathbf{x}_0\|^2}{\sqrt{D_a^{-1} + \|\Delta \mathbf{x}_0 \cdot \mathbf{U}_a\|^2 + \|\Delta \mathbf{x}_0 \cdot \mathbf{U}_b\|^2}}. \tag{A.18}$$

The corresponding loss is given by

$$\mathcal{L}_{\text{no-bias}} = \sum_{(a,b)} \exp\left(-\alpha\, \mathrm{SNR}_{a,b}^{\text{no-bias}}\right). \tag{A.19}$$

No-dimensionality (No-D) ablation. To assess the role of manifold dimensionality, the dimensionality term was removed from the SNR:

$$\mathrm{SNR}_{a,b}^{\text{no-D}} = \frac{\|\Delta \mathbf{x}_0\|^2 + \left(\frac{R_b^2}{R_a^2} - 1\right)}{\sqrt{\|\Delta \mathbf{x}_0 \cdot \mathbf{U}_a\|^2 + \|\Delta \mathbf{x}_0 \cdot \mathbf{U}_b\|^2 + \varepsilon}}, \tag{A.20}$$

where ε is a small constant for numerical stability and is set to 0.1. The corresponding loss is given by

$$\mathcal{L}_{\text{no-}D} = \sum_{(a,b)} \exp\left(-\alpha\, \mathrm{SNR}_{a,b}^{\text{no-}D}\right). \tag{A.21}$$

No-overlap ablation. To evaluate the contribution of the signal–noise overlap term, it was removed from the SNR:

$$\mathrm{SNR}_{a,b}^{\text{no-overlap}} = \frac{\|\Delta \mathbf{x}_0\|^2 + \left(\frac{R_b^2}{R_a^2} - 1\right)}{\sqrt{D_a^{-1}}}. \tag{A.22}$$

The corresponding loss is given by

$$\mathcal{L}_{\text{no-overlap}} = \sum_{(a,b)} \exp\left(-\alpha\, \mathrm{SNR}_{a,b}^{\text{no-overlap}}\right). \tag{A.23}$$

Training protocol. All ablation variants were trained under identical conditions to the full SNR loss, ensuring a controlled comparison. Specifically, the same network architecture, optimizer, learning rate schedule, batch construction, number of training steps, and all other hyperparameters were used across all ablations. For all ablated losses, the exponential scaling parameter was fixed to $\alpha = 0.5$, matching the value used in the full SNR objective.

Ablation results. Table A.10 summarizes the effects of selectively removing individual components of the SNR loss. The full SNR loss achieves the lowest generalization error and the highest generalization SNR. Optimizing the signal alone (signal-only) increases signal but induces severe dimensionality reduction, leading to reduced SNR and poor generalization. Removing the signal term (no-signal) catastrophically impairs performance: centroid separation vanishes, SNR collapses, and dimensionality explodes. This confirms that signal is necessary for generalization. Removing the bias term degrades generalization and reduces SNR, primarily by weakening the signal. Removing the dimensionality term (no-D) produces a modest performance drop. Although dimensionality decreases, signal and generalization SNR remain relatively high. The no-overlap ablation increases the signal but sharply reduces dimensionality and generalization performance. Collectively, these ablations provide causal evidence that relational generalization depends on a finely tuned geometric balance. Each SNR term must be jointly regularized. The full SNR loss uniquely enforces this

balance, yielding representations that best support robust relational generalization.

A.11.7 Extension of the SNR loss

Although the SNR loss of Eq. 2.11 is defined as a uniform average over ordered rule pairs, this choice is not restrictive. Rule pairs can be reweighted to emphasize underrepresented, difficult, or task-relevant relations. Because the objective decomposes over pairwise geometric terms, such weighting or adaptive sampling extensions preserve the underlying formulation and its geometric interpretation.

As an illustrative example, we introduce a relation-weighted variant of the SNR loss. Rather than enforcing uniform contributions across all ordered rule pairs, a fixed weight function w_a is assigned to each pair (a, b) with target rule a:

$$\mathcal{L}_{\mathrm{SNR}}^{\mathrm{weighted}} = \sum_{(a,b)} w_a \, \exp\left(-\alpha \mathrm{SNR}_{a,b}\right) . \tag{A.24}$$

The geometric term $\mathrm{SNR}_{a,b}$ is computed identically to the unweighted loss.

For example, we assigned normalized weights of 0.6, 1.6, and 1.1 to Progression, Arithmetic, and Logical rules, respectively. Models were trained with the relation-weighted SNR loss using the same architecture, optimizer, batch construction, and hyperparameters as in the standard (uniform) SNR loss experiments. Table A.11 reports generalization performance and geometric metrics for both unweighted and relation-weighted SNR training. Notably, relation-weighted SNR improves generalization on Arithmetic and Logical rules relative to the uniform SNR loss, while leaving overall performance and the underlying geometric structure largely unchanged. This experiment demonstrates that such modifications work in practice and showcases the flexibility of the SNR objective. Because the weighting enters the loss in a modular fashion, it can be readily extended to more sophisticated strategies, such as adaptive weighting or sampling based on rule-wise performance estimated online

during training.

Table A.8: Generalization error and geometric quantities with respect to the exponential weighting parameter α in the SNR loss (mean $\pm$ s.e.m.).

α	Error	Signal	D	SNR
0.1	0.263 ± 0.007	0.39 ± 0.02	34.27 ± 1.77	0.62 ± 0.03
0.5	0.096 ± 0.008	3.31 ± 0.34	14.04 ± 0.74	2.04 ± 0.11
1.0	0.162 ± 0.020	3.45 ± 0.76	13.50 ± 0.95	1.84 ± 0.30
2.0	0.159 ± 0.018	6.45 ± 0.94	9.21 ± 0.53	1.87 ± 0.23

Table A.9: Comparison of SNR loss, VICReg, cross-entropy (CE), and Prototypical objectives. Values are mean $\pm$ s.e.m. across 15 held-out rule splits.

Objective	**Gen. Error**	**Gen. SNR**	**Gen. Signal**	**Gen.** D
SNRloss	0.096 ± 0.008	2.04 ± 0.11	3.31 ± 0.34	14.04 ± 0.74
VICReg	0.126 ± 0.009	1.95 ± 0.12	4.44 ± 0.48	12.12 ± 0.42
CE	0.153 ± 0.018	2.03 ± 0.25	5.15 ± 0.72	9.15 ± 0.38
Prototypical	0.228 ± 0.011	0.85 ± 0.05	0.78 ± 0.07	23.08 ± 1.17

Table A.10: Ablation analysis of the SNR loss. Values are mean $\pm$ s.e.m. across five random splits for generalization error, estimated generalization SNR, centroid separation (signal), and manifold dimensionality D, all measured on held-out rules.

Loss variant	**Gen. Error**	**Gen. SNR**	**Gen. Signal**	**Gen.** D
Full SNR loss	0.096 ± 0.009	2.04 ± 0.12	3.31 ± 0.26	14.04 ± 0.90
Signal-only	0.174 ± 0.014	1.15 ± 0.07	3.39 ± 0.45	4.96 ± 0.17
No-signal	0.492 ± 0.004	0.03 ± 0.01	0.00 ± 0.00	272.87 ± 3.86
No-bias	0.148 ± 0.021	1.75 ± 0.29	2.78 ± 0.69	12.20 ± 0.42
No-D	0.112 ± 0.016	1.94 ± 0.24	3.02 ± 0.45	11.11 ± 1.24
No-overlap	0.134 ± 0.025	1.46 ± 0.15	4.26 ± 0.79	6.51 ± 0.16

Table A.11: Effect of relation-weighted SNR loss on generalization performance. Values are reported as mean $\pm$ s.e.m.

Loss	**Overall Error**	**Prog. Error**	**Arith. Error**	**Logical Error**
SNR (uniform)	0.096 ± 0.008	0.039 ± 0.008	0.111 ± 0.019	0.154 ± 0.011
SNR (weighted)	0.101 ± 0.009	0.060 ± 0.011	0.106 ± 0.020	0.147 ± 0.013

B

Supporting Material for Chapter 3

This appendix collects supporting material for Chapter 3.

B.1 Stimulus sets

B.1.1 Construction of the word set

The word-level analyses of Chapter 3 use LibriSpeech [100], a corpus of English audiobook speech read by volunteers, segmented into utterances of at most 35 s with a reference transcript. Stimuli were drawn from the `test-clean` (2,620 utterances, 40 readers) and `train-clean-360` (104,014 utterances, 921 readers) partitions, which were pooled to yield more occurrences per word type. Below, these utterances are referred to as sentences, although they need not correspond to complete grammatical sentences. Word boundaries are taken from the LibriSpeech Alignments dataset [101,111], word and phoneme alignments of the reference transcripts generated with the Montreal Forced Aligner [102] and its pretrained LibriSpeech acoustic model, which gives each word an onset and an offset on a 10 ms grid.

The unit of analysis is one spoken occurrence of a word within one utterance, and a word manifold is a set of such occurrences of the same word, drawn from distinct utterances and spanning variation in reader, speaking rate, and sentence context. An occurrence was eligible only if the word began at least 5 s after the utterance onset and ended within its first 30 s, so that in the in-sentence condition of Section 3.3 every word is preceded by at least 5 s of context and the audio from utterance onset to word offset fits Whisper's 30 s input window. This rule, not the exclusions below, is the binding constraint on the vocabulary: it removes every utterance-initial occurrence and roughly halves the number of word types that reach the occurrence threshold. Estimating a manifold's geometry requires many occurrences per word, while the pair-level analyses of Sections 3.6 and 3.8 require many words, and because word frequency is heavily skewed, these pull in opposite directions. Two hundred occurrences per word is the compromise: it is enough for a 200-point sample to carry any dimensionality the chapter reports, since the covariance of n points has rank at most $n - 1$, and it leaves 195 held-out points per word for the $m = 5$ readout, while still admitting close to a

thousand word types.

Each word was represented by 200 eligible occurrences drawn from distinct utterances, so that no two points of a manifold share a sentence, sampled uniformly without replacement with a fixed seed; word types with fewer than 200 eligible occurrences in distinct utterances were dropped, leaving a pool of 966 types.

Three exclusions were then applied, each removing words for which the analyses of the chapter would not measure what they are meant to measure. First, five words for which no single Whisper vocabulary token decoded to the complete word were removed, because they do not satisfy the whole-word decoder readout convention of Section 3.3: every retained word is rendered by a single whole-word token, although several token identities may encode surface variants of the same word, and the state that predicts that token is the decoder representation of the word. Removing them from both stacks keeps one vocabulary throughout. Second, function words were removed by part-of-speech tag, assigned to the word in isolation, because determiners, prepositions, conjunctions, pronouns and modals are short, phonetically reduced, and largely predictable from syntax, so their manifolds would reflect reduction and context rather than lexical identity, and their frequency would otherwise let them dominate the pool (102). Third, a fixed list removed what the tagger cannot judge from the word form alone: closed-class words that it labels as verbs, adverbs or adjectives in isolation, among them the copulas *are* and *am*, the archaic pronouns *thy* and *ye*, and discourse adverbs and quantifiers such as *so*, *only* and *just*; together with numerals and proper names, whose identity is not lexical meaning of the kind probed in Section 3.6 (59). The result is a vocabulary of 800 word types and 160,000 tokens, drawn from 75,112 utterances and 961 readers. By isolated tag, it is 63% nouns, 18% verbs, 10% adjectives and 8% adverbs. A few closed-class words remain: the auxiliaries *had*, *have*, *did* and *got*, which tag as verbs, and *there*, *beneath*, *beside* and *towards*, which take a function tag in most of their sentences; results on the effects of sentence context in Section 3.8 should therefore be read as applying primarily, but not exclusively, to content

words. The 133 closed-class words removed at the last two steps retain their 200 sampled occurrences in the released stimulus files, so that they can be analyzed as a separate class. Selected words have a median duration of 380 ms (interquartile range 300–480), or about 19 encoder frames, and 1 to 12 phonemes (median 4). The word list, the sampled occurrences with their timestamps, and the exclusion audit are released with the analysis code.

B.1.2 The phoneme set

Table B.1 lists the 39 phoneme labels of the phoneme-level analyses, the TIMIT transcription symbols folded into each, the distinctive features used in Section 3.7, and the counts behind the sampling rule of Section 3.10.5. The reduction is applied to the phonetic transcriptions of all 6,300 TIMIT sentences (241,225 labeled intervals). The four non-speech symbols, utterance-initial and -final silence `h#`, pause `pau`, epenthetic silence `epi`, and the glottal stop `q`, are removed (20,777 intervals). The remaining 57 symbols are mapped onto the 39 labels. Adjacent intervals carrying the same label are then merged (22,791 merges, almost all a stop closure joined to its release), leaving 197,657 occurrences. The labels are listed in the order of the matrix figures of Section 3.7, grouped by manner of articulation; the grouping is for reading only, and the analyses treat all 39 alike.

Occurrence counts are unequal by more than an order of magnitude, from 225 for /ZH/ to 18,272 for /IH/, and the sampling rule equalizes them: each manifold is 200 occurrences from 200 distinct sentences, so that no two points of a manifold share a recording. Every label has enough distinct sentences for the rule (219 for /ZH/, at least 689 for every other label), and the 200 selected occurrences of each label come from between 162 and 183 of the 630 speakers. The selected occurrences span the corpus: 27% come from the `TEST` partition and 73% from `TRAIN`, in proportion to their sizes, and 18% come from the two dialect sentences (`SA1`, `SA2`) that every TIMIT speaker reads, so a minority of each manifold's points share their lexical context with other points while differing in speaker. Median phone duration over all occurrences ranges from 34 ms (/DH/) to 159 ms

(/OY/); over the selected occurrences it is 82 ms (interquartile range 55–115), which under the pooling rule of Section 3.10.5 gives a median of five encoder positions per occurrence (range 1–23). The label map, the selected occurrences with their sample boundaries and the reduction audit are released with the analysis code.

Table B.1: The phoneme inventory. For each of the 39 labels: the TIMIT transcription symbols mapped onto it (closures such as `bcl` merge with an adjacent release into one occurrence), its distinctive features (manner, place and voicing for consonants; height, backness, rounding, tenseness and diphthong status for vowels), the number of occurrences after reduction and merging in all 6,300 sentences, the number of distinct sentences containing the label, and the number of speakers among the 200 selected occurrences. The last row gives the totals and the removed symbols.

Label	TIMIT symbols	Features	Occurrences	Utterances	Speakers
Stops					
/P/	`p`, `pcl`	stop, labial, voiceless	4,014	2,696	181
/B/	`b`, `bcl`	stop, labial, voiced	3,405	2,454	176
/T/	`t`, `tcl`	stop, alveolar, voiceless	9,847	5,026	167
/D/	`d`, `dcl`, `dx`	stop, alveolar, voiced	11,481	5,341	162
/K/	`k`, `kcl`	stop, velar, voiceless	8,270	4,654	170
/G/	`g`, `gcl`	stop, velar, voiced	3,277	2,924	179
Affricates					
/CH/	`ch`	affricate, postalveolar, voiceless	1,081	943	176
/JH/	`jh`	affricate, postalveolar, voiced	1,581	1,343	176
Fricatives					
/F/	`f`	fricative, labial, voiceless	3,128	2,281	182
/V/	`v`	fricative, labial, voiced	2,704	2,063	176
/TH/	`th`	fricative, dental, voiceless	1,018	912	174
/DH/	`dh`	fricative, dental, voiced	3,879	3,119	173
/S/	`s`	fricative, alveolar, voiceless	10,114	5,191	172
/Z/	`z`	fricative, alveolar, voiced	5,046	3,305	173
/SH/	`sh`	fricative, postalveolar, voiceless	3,034	2,081	177
/ZH/	`zh`	fricative, postalveolar, voiced	225	219	173
/HH/	`hh`, `hv`	fricative, glottal, voiceless	2,835	2,332	173
Nasals					
/M/	`m`, `em`	nasal, labial, voiced	5,600	3,719	171
/N/	`n`, `en`, `nx`	nasal, alveolar, voiced	11,869	5,367	172
/NG/	`ng`, `eng`	nasal, velar, voiced	1,787	1,556	170
Approximants					
/L/	`l`, `el`	lateral, alveolar, voiced	9,413	5,179	177
/R/	`r`	rhotic, postalveolar, voiced	9,064	4,854	183
/W/	`w`	glide, labial, voiced	4,379	2,873	169
/Y/	`y`	glide, palatal, voiced	2,349	1,822	176
Monophthongs					
/IY/	`iy`	high, front, unrounded, tense	9,660	4,868	173
/IH/	`ih`, `ix`	high, front, unrounded, lax	18,272	5,893	168
/EY/	`ey`	mid, front, unrounded, tense	3,087	2,304	172
/EH/	`eh`	mid, front, unrounded, lax	5,293	3,648	176
/AE/	`ae`	low, front, unrounded, lax	5,404	3,349	176
/AH/	`ah`, `ax`, `ax-h`	mid, central, unrounded, lax	8,634	4,160	178
/ER/	`er`, `axr`	mid, central, unrounded, tense	7,610	4,082	173
/AA/	`aa`	low, back, unrounded, tense	4,197	2,927	170
/AO/	`ao`	low, back, rounded, tense	4,096	2,731	174
/OW/	`ow`	mid, back, rounded, tense	2,910	2,444	178
/UH/	`uh`	high, back, rounded, lax	756	689	176
/UW/	`uw`, `ux`	high, back, rounded, tense	3,211	2,693	176
Diphthongs					
/AY/	`ay`	low, front, unrounded, tense, diphthong	3,235	2,688	180
/AW/	`aw`	low, back, unrounded, tense, diphthong	945	853	180
/OY/	`oy`	mid, back, rounded, tense, diphthong	947	910	179
All labels (removed: `h#`, `pau`, `epi`, `q`)			197,657	6,300	630

B.2 Layer indexing

Table B.2 lists the 67 stages of the layer axis defined in Section 3.3 and, for each, the module of the `openai-whisper` implementation whose output is read. The rules for reading them are given in Section 3.10.2. The log-mel spectrogram has no index, because it is the input to the network rather than a stage of its computation.

Table B.2: Layer indices and the modules they read. k is the layer index used throughout Chapter 3. The module column names the attribute of `model.encoder` or `model.decoder` whose output is read; `blocks[`j`]` denotes the residual stream returned by the block, after all of its residual additions.

Encoder (`model.encoder`)	
k	Module whose output is read
0	`conv2` (after GELU) + `positional_embedding`
1	`blocks[0]`
2	`blocks[1]`
3	`blocks[2]`
4	`blocks[3]`
5	`blocks[4]`
6	`blocks[5]`
7	`blocks[6]`
8	`blocks[7]`
9	`blocks[8]`
10	`blocks[9]`
11	`blocks[10]`
12	`blocks[11]`
13	`blocks[12]`
14	`blocks[13]`
15	`blocks[14]`
16	`blocks[15]`
17	`blocks[16]`
18	`blocks[17]`
19	`blocks[18]`
20	`blocks[19]`
21	`blocks[20]`
22	`blocks[21]`
23	`blocks[22]`
24	`blocks[23]`
25	`blocks[24]`
26	`blocks[25]`
27	`blocks[26]`
28	`blocks[27]`
29	`blocks[28]`
30	`blocks[29]`
31	`blocks[30]`
32	`blocks[31]`
33	`ln_post`

Decoder (`model.decoder`)	
k	Module whose output is read
34	`blocks[0]`
35	`blocks[1]`
36	`blocks[2]`
37	`blocks[3]`
38	`blocks[4]`
39	`blocks[5]`
40	`blocks[6]`
41	`blocks[7]`
42	`blocks[8]`
43	`blocks[9]`
44	`blocks[10]`
45	`blocks[11]`
46	`blocks[12]`
47	`blocks[13]`
48	`blocks[14]`
49	`blocks[15]`
50	`blocks[16]`
51	`blocks[17]`
52	`blocks[18]`
53	`blocks[19]`
54	`blocks[20]`
55	`blocks[21]`
56	`blocks[22]`
57	`blocks[23]`
58	`blocks[24]`
59	`blocks[25]`
60	`blocks[26]`
61	`blocks[27]`
62	`blocks[28]`
63	`blocks[29]`
64	`blocks[30]`
65	`blocks[31]`
66	`ln`

B.3 The prototype signal-to-noise estimator

Chapter 3 applies the geometric framework of Chapter 2 to the same m-shot prototype readout, using $m = 5$ and retaining the full second-order variance expression given below.

The readout is the few-shot word classification of Figure 3.3a. A held-out utterance of word a is classified between words a and b from m example utterances of each: the examples are averaged into prototypes $\bar{\mathbf{x}}_a$ and $\bar{\mathbf{x}}_b$, and the utterance is assigned to the word whose prototype is nearer in representation space, with chance error 0.5. Each utterance is its word's centroid plus a within-word deviation. The test utterance is $\xi = \mathbf{x}_0^a + \eta$, the prototypes are $\bar{\mathbf{x}}_a = \mathbf{x}_0^a + \bar{\eta}_a$ and $\bar{\mathbf{x}}_b = \mathbf{x}_0^b + \bar{\eta}_b$ with $\bar{\eta}_a$ and $\bar{\eta}_b$ the means of m independent deviations that are also independent of the test deviation η, and $\Delta = \mathbf{x}_0^a - \mathbf{x}_0^b$ separates the two word centroids. The decision variable $h = \|\xi - \bar{\mathbf{x}}_b\|^2 - \|\xi - \bar{\mathbf{x}}_a\|^2$ is positive on average, and under the Gaussian approximation of Section 2.5.2 the error probability is $H(\mathbb{E}\, h / \sqrt{\mathrm{Var}\, h})$. In the normalized notation of Chapter 2,

$$\mathrm{SNR}_m = \frac{1}{2} \frac{\|\Delta \mathbf{x}_0\|^2 + \left(R_b^2/R_a^2 - 1\right)/m}{\sqrt{D_a^{-1}/m + \|\Delta \mathbf{x}_0 \cdot \mathbf{U}_b\|^2/m + \|\Delta \mathbf{x}_0 \cdot \mathbf{U}_a\|^2 + \|\mathbf{U}_a^\top \mathbf{U}_b\|_F^2/m + \beta_{ab}/m^2}}, \quad \text{(B.1)}$$

where $\beta_{ab} = \frac{1}{2}\left(D_a^{-1} + (R_b^4/R_a^4)\, D_b^{-1}\right)$, $\Delta \mathbf{x}_0 = \Delta / R_a$ as in Eq. 2.6, $\mathbf{U}_a$ and $\mathbf{U}_b$ are the radius-scaled principal axes of Eq. 2.7, each radius in units of R_a, and $\|\mathbf{U}_a^\top \mathbf{U}_b\|_F$ is the Frobenius norm of the matrix of their inner products. The last term of the denominator, β_{ab}/m^2, is the bias–variance term and the preceding one the noise–noise overlap term; both are absent from the dominant-term forms of Eqs. 2.7 and A.4, which the remaining terms reproduce exactly at $m = 1$ and at general m. Here $R_q^2 = \mathrm{tr}\, \Sigma_q$ and $D_q = (\mathrm{tr}\, \Sigma_q)^2 / \mathrm{tr}(\Sigma_q^2)$ denote the squared manifold radius and the participation-ratio dimensionality for $q \in \{a, b\}$.

B.3.1 The prediction at every layer

Figure 3.4 shows the measured five-shot error of every ordered word pair against its signal-to-noise ratio at six stages. Figures B.1 and B.2 show the same plot at every stage: the log-mel input and the 34 encoder layers, and the 33 decoder layers. Each panel draws all 800×799 pairs against the prediction $\varepsilon = H(\mathrm{SNR}_{m=5})$ of Eq. 2.8. Through the encoder, the cloud of pairs slides down the curve from one layer to the next as the mean ratio in Figure 3.5b rises. In the decoder, the cloud stays in place through layer 54 and then moves down the curve over the last twelve layers. At decoder layers 35 to 43, a separate group of pairs lies below the curve, at ratios near 2 with error near zero, and is classified better than its ratio predicts; the group is absent from layer 44 onward.

B.3.2 The three remaining terms

Section 3.5 follows the signal, dimensionality, and signal–noise overlap term across depth and states that the other three terms of Eq. B.1 are small. Figure B.3 shows them. The bias term is below 0.016 at every layer and never exceeds 2% of the numerator $\|\Delta \mathbf{x}_0\|^2 + \text{bias}$. The bias–variance term is below 0.007 and never exceeds 6% of the sum of the four noise terms. The noise–noise overlap term is below 0.022; its share of the noise sum is largest at decoder layers 34 to 38, where it reaches 22%, and is below 20% at every other layer. None of the three reverses the direction of any change in the ratio reported in Section 3.5.

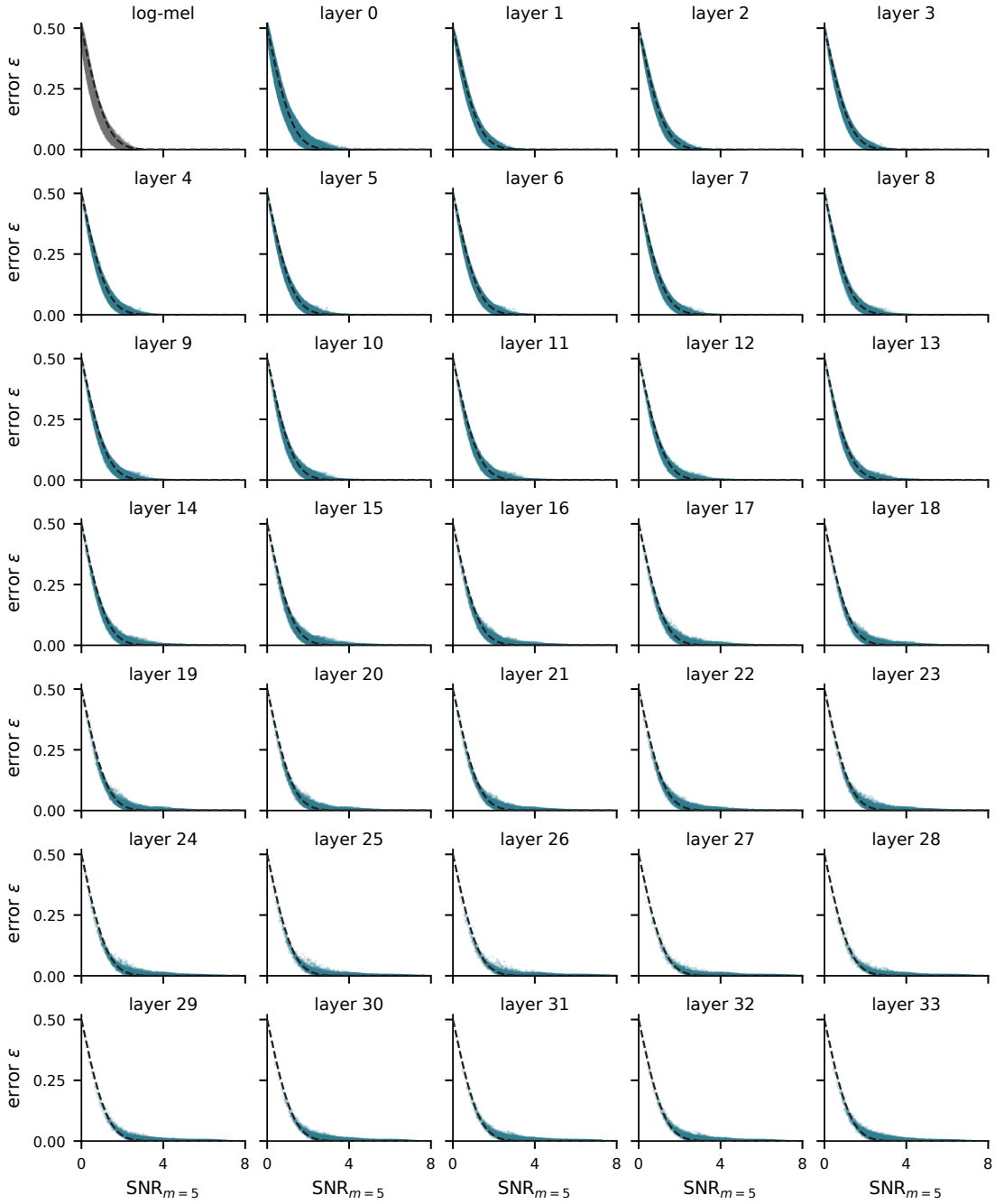


Figure B.1: Few-shot error against signal-to-noise ratio at every encoder stage, single-word condition. As Figure 3.4, for the log-mel input and encoder layers 0 to 33: each panel shows all 800×799 ordered pairs, and the dashed curve is $\varepsilon = H(\mathrm{SNR}_{m=5})$ of Eq. 2.8.

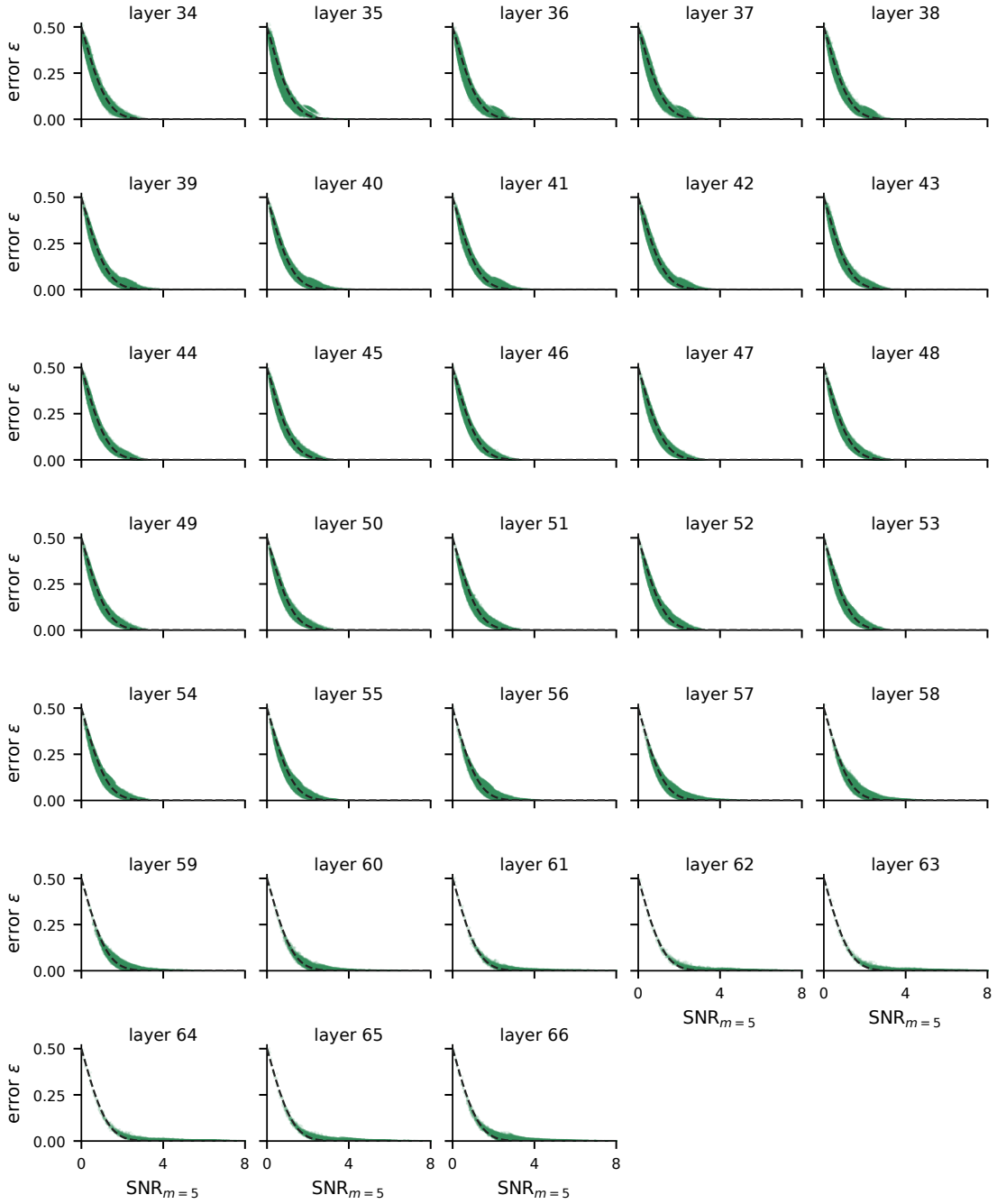


Figure B.2: Few-shot error against signal-to-noise ratio at every decoder stage, single-word condition. As Figure 3.4, for decoder layers 34 to 66: each panel shows all 800×799 ordered pairs, and the dashed curve is $\varepsilon = H(\mathrm{SNR}_{m=5})$ of Eq. 2.8.

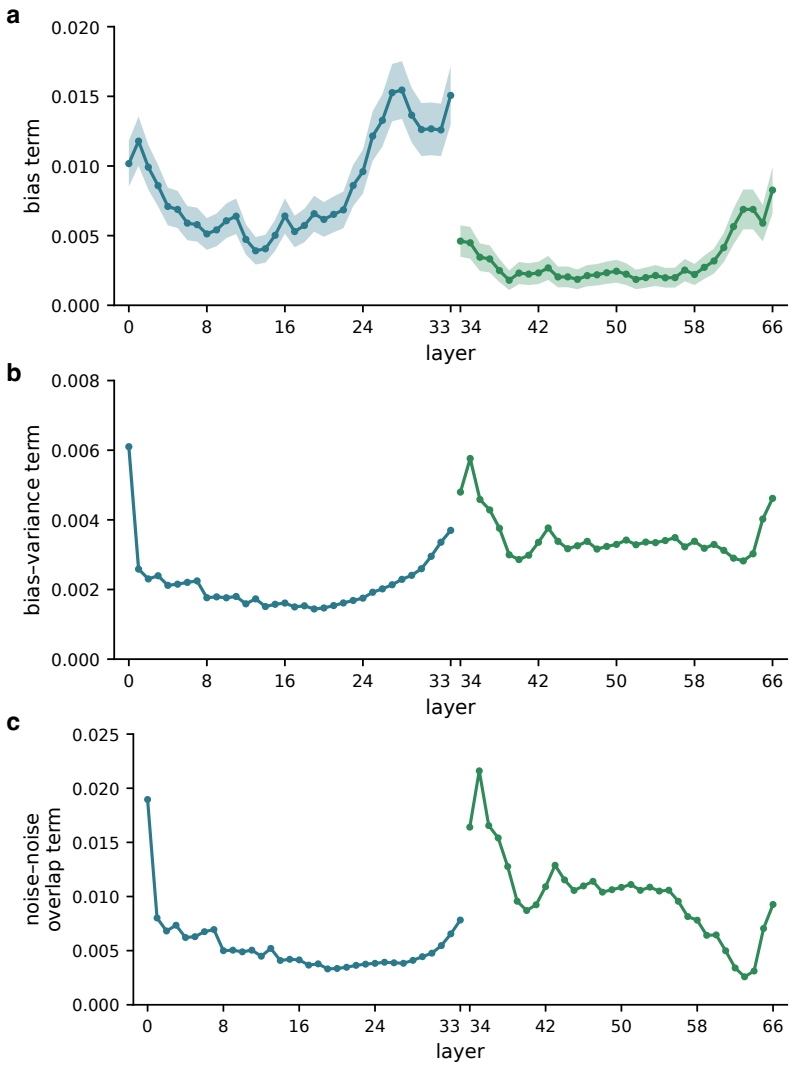


Figure B.3: The three terms of Eq. B.1 not followed in Section 3.5, single-word condition. (a) The bias term $(R_b^2/R_a^2 - 1)/m$. (b) The bias–variance term β_{ab}/m^2. (c) The noise–noise overlap term $\|\mathbf{U}_a^\top \mathbf{U}_b\|_F^2/m$ in the normalization of Eq. B.1. Encoder in teal (layers 0–33), decoder in green (layers 34–66); $m = 5$. Lines are means over all 800×799 ordered pairs and bands are $\pm$ one s.e.m. across the 800 words. The band in (a) is wide relative to the mean because the bias of the pair (a, b) is close to the negative of the bias of (b, a), so the per-word means scatter about a small value.

B.4 Supplementary analyses of linguistic features

This appendix reports, for the matched pair groups of Section 3.6, the signal term of the signal-to-noise ratio for the three target measures; the ratio for the input acoustic distance as a target; and two checks on the semantic result: a second semantic measure and grammatical class. The groups, the bootstrap, and the layer-wise test are those of Section 3.10.7.

B.4.1 The signal term of the matched groups

Figure B.4 shows the signal term, the squared separation between two word centroids relative to the manifold radius, for the large- and small-difference groups of Figures 3.7 and 3.8a, single-word condition. For the duration difference, the large-difference group carries the larger signal at every layer from 0 to 60: 2.74 against 0.37 at layer 0, 3.39 against 3.06 at the encoder peak, and 3.54 against 2.58 at the encoder output; the two groups are nearly equal at the decoder peak, 3.42 against 3.36. For phoneme-sequence distance, the two groups' signals lie close together through layer 14. The large-distance group has a larger signal from layer 15 to the encoder output (3.55 vs 2.38 at layer 27) and from layer 56 to the decoder output (4.19 vs 2.67 at layer 63). For the semantic distance, the diagonal follows the ratio: the semantically distant group has the larger signal from layer 25 onward, 3.11 against 2.92 at the encoder peak and 3.40 against 3.21 at the decoder peak.

B.4.2 Input acoustic distance

The signal term computed on the 128-dimensional log-mel spectrogram measures how far apart two words' input representations lie relative to their spread. Over all 319,600 pairs, it is uncorrelated with the phoneme-sequence distance and with the duration difference (Spearman rank correlation 0.01 for each), because the pooled spectrogram discards the order of the frames. As a target, it was matched on the three measures of the chapter and on the phoneme-count difference; the groups

hold 15,729 pairs each, the mean input distance is 2.72 versus 0.11, and the standardized mean differences for the controls are at most 0.09. Pairs that are far apart at the input are more separable throughout the encoder, most in its early layers (Figure B.5): 1.39 against 0.64 at layer 0, 3.02 against 1.54 at layer 9, 5.93 against 5.44 at the encoder peak, and 4.92 against 4.19 at the encoder output, significant at every encoder layer. Along the decoder plateau, the two groups coincide, and a smaller advantage returns from layer 59, 6.18 against 5.81 at the decoder peak.

B.4.3 A second semantic measure

We first analyzed Section 3.6 using distances between context-free BERT embeddings (Section 3.10.7), which correlate weakly with the word2vec distances (Spearman rank correlation 0.16). Matched on the same controls as the word2vec comparison, in the same 623 cells and with 13,954 pairs per group (mean distance 0.65 against 0.22; standardized mean differences of the controls at most 0.03), the BERT distance separates the groups at no layer in the single-word condition, the largest difference being 0.19 at layer 59 (Figure B.6a). In the in-sentence condition, it separates them only along the decoder plateau, significantly over layers 50 to 60, 1.81 against 1.47 at layer 51 and 3.03 against 2.64 at layer 59, with no difference at the decoder peak, 6.65 against 6.61 (Figure B.6b).

B.4.4 Grammatical class

Of the 800 words, 507 are tagged as nouns, 141 as verbs, 80 as adjectives, and 60 as adverbs when tagged in isolation. Pairs of words of different class were compared with pairs of the same class, matched on duration difference, phoneme-sequence distance, phoneme-count difference, input acoustic distance, and the pair's difference in and mean of log frequency (601 cells, 113,153 pairs per group; standardized mean differences of the controls at most 0.05). Different-class pairs are slightly more separable along the decoder plateau in both conditions (Figure B.7): in the single-word condi-

tion, significantly over layers 32 to 61, 1.20 against 1.04 at layer 51 and 2.82 against 2.62 at layer 59, with no difference at the decoder peak, 6.07 against 6.05; in the in-sentence condition, significantly over layers 29 to 64, 1.76 against 1.56 at layer 51 and 3.02 against 2.75 at layer 59, 6.73 against 6.60 at the peak and 5.90 against 5.89 at the output.

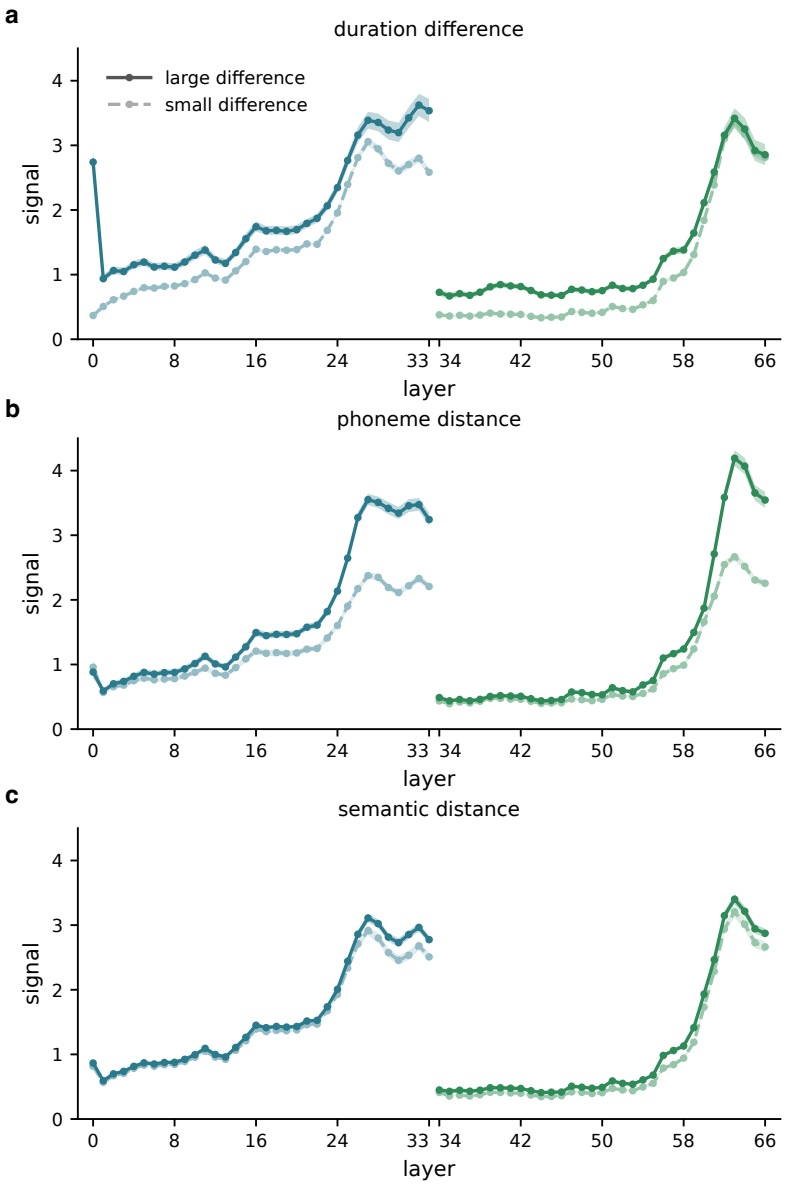


Figure B.4: Signal term of the matched pair groups, single-word condition. The signal term of Eq. 2.6 against layer for the large-difference (solid) and small-difference (dashed, lighter) groups of Figures 3.7 and 3.8a: (a) duration difference, (b) phoneme-sequence distance, (c) semantic distance. Encoder in teal (layers 0–33), decoder in green (layers 34–66). Bands are the 2.5th to 97.5th percentiles of a bootstrap over the 800 word types (10,000 draws).

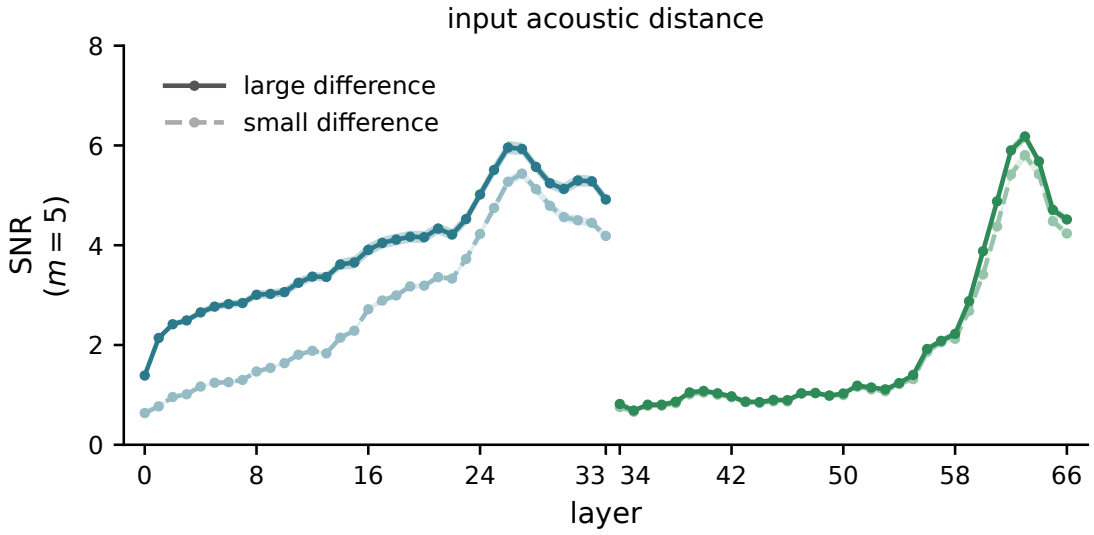


Figure B.5: Input acoustic distance as a target, single-word condition. Mean $\mathrm{SNR}_{m=5}$ against layer for the large-difference (solid) and small-difference (dashed, lighter) groups in the signal term on the log-mel spectrogram ($15{,}729$ pairs per group, mean 2.72 against 0.11), matched on duration difference, phoneme-sequence distance, semantic distance and phoneme-count difference. Encoder in teal (layers 0–33), decoder in green (layers 34–66). Bands are the 2.5th to 97.5th percentiles of a bootstrap over the 800 word types ($10{,}000$ draws).

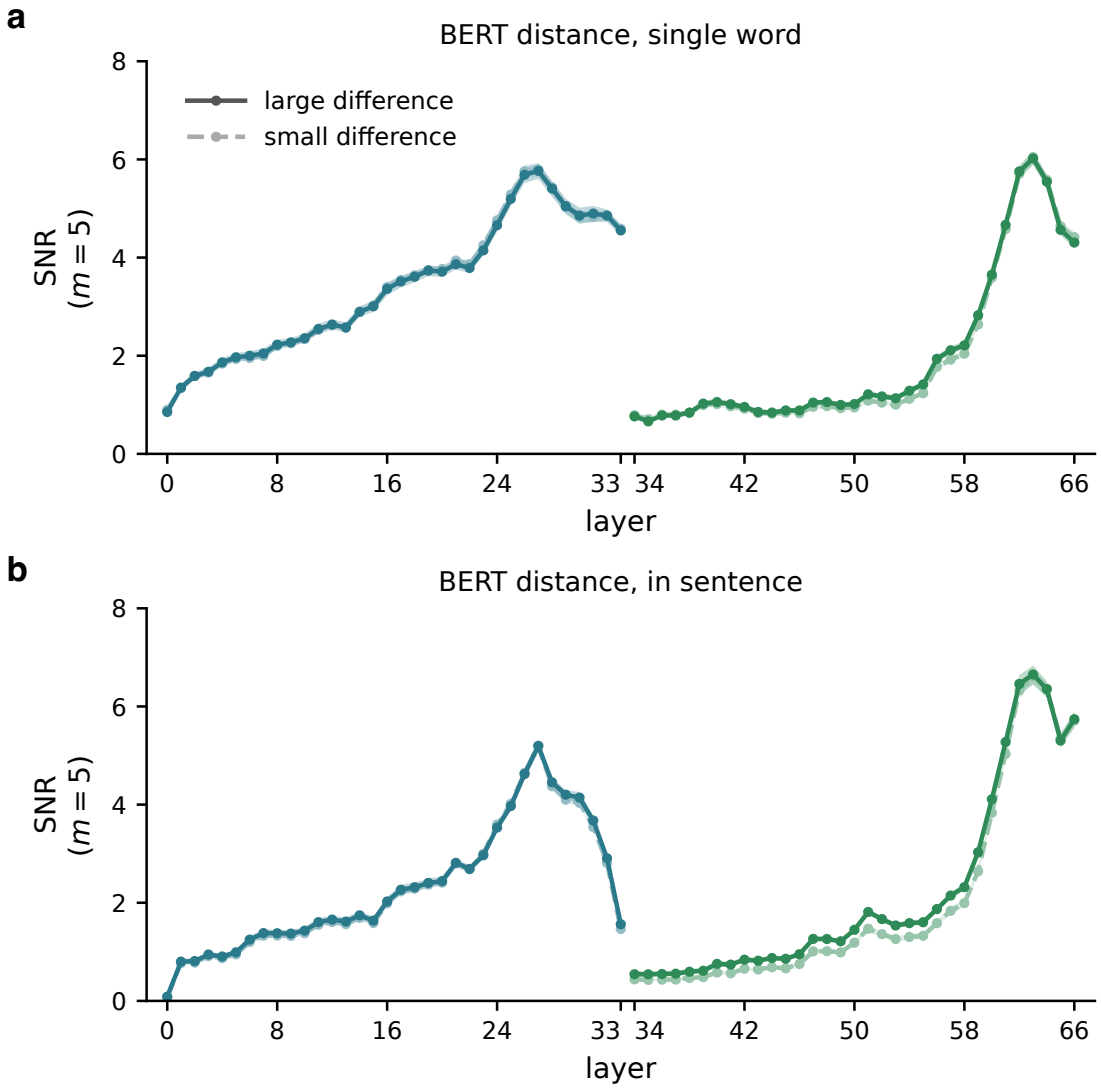


Figure B.6: Distance between context-free BERT embeddings as the semantic measure. Mean $\mathrm{SNR}_{m=5}$ against layer for the large-distance (solid) and small-distance (dashed, lighter) groups, $13{,}954$ pairs per group, matched as the word2vec groups of Figure 3.8, in (a) the single-word and (b) the in-sentence condition. Encoder in teal (layers 0–33), decoder in green (layers 34–66). Bands are the 2.5th to 97.5th percentiles of a bootstrap over the 800 word types ($10{,}000$ draws).

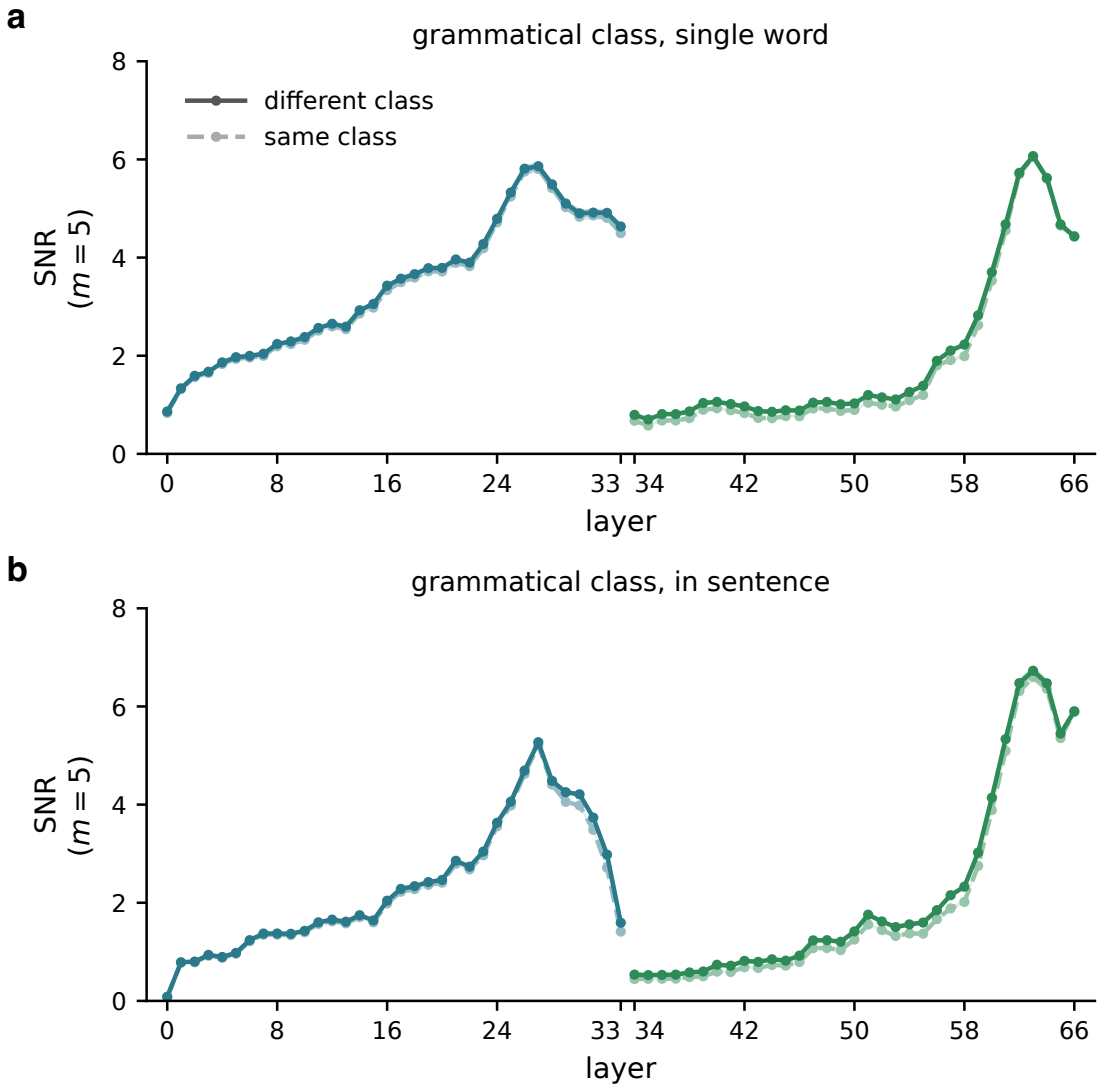


Figure B.7: Grammatical class. Mean $\mathrm{SNR}_{m=5}$ against layer for pairs of words of different grammatical class (solid) and pairs of the same class (dashed, lighter), 113,153 pairs per group, matched on duration difference, phoneme-sequence distance, phoneme-count difference, input acoustic distance and log frequency, in (a) the single-word and (b) the in-sentence condition. Encoder in teal (layers 0–33), decoder in green (layers 34–66). Bands are the 2.5th to 97.5th percentiles of a bootstrap over the 800 word types (10,000 draws).

B.5 Supplementary analyses of phoneme manifolds

B.5.1 Phoneme-pair separability with every phoneme labeled

Figures B.8, B.9 and B.10 show the three matrices of Figure 3.11 at full size, with every phoneme labeled.

B.5.2 Phoneme-pair separability at every encoder layer

Figure B.11 shows the pair matrix of Figure 3.11 at the log-mel input and at every encoder layer. The ratio of within-class to between-class separability rises through the encoder to layer 27, dips over layers 28 to 31 and recovers at the output: 0.47 on the spectrogram, 0.56 at layer 9, 0.69 at layer 18, 0.79 at layer 27, 0.75 at layers 30 and 31, and 0.80 at layer 33.

B.5.3 Prototype error against the theory at every encoder layer

Figure B.12 shows the measured five-shot error of every ordered phoneme pair against its signal-to-noise ratio, with the prediction $\varepsilon = H(\mathrm{SNR})$ of Eq. 2.8, at the log-mel input and at every encoder layer; the four stages of Figure 3.9b are among them. At every layer from 1 to 33, the correlation between measured and predicted error over the 1,482 pairs is at least 0.98, and the mean measured and predicted errors differ by at most 0.018. Layer 0, the output of the convolutional stem, is the one stage at which the points lie above the curve at low ratio, with a correlation of 0.97.

B.5.4 Margin of the 39-way linear probe

Figure B.13 shows the margin of the 39-way linear probe whose accuracy is reported in Section 3.7. The margin increases from 1.9 at layer 0 to 4.4 at layer 21 and 6.2 at the encoder output, through the layers in which the pairwise signal-to-noise ratio falls from 1.89 to 0.75.

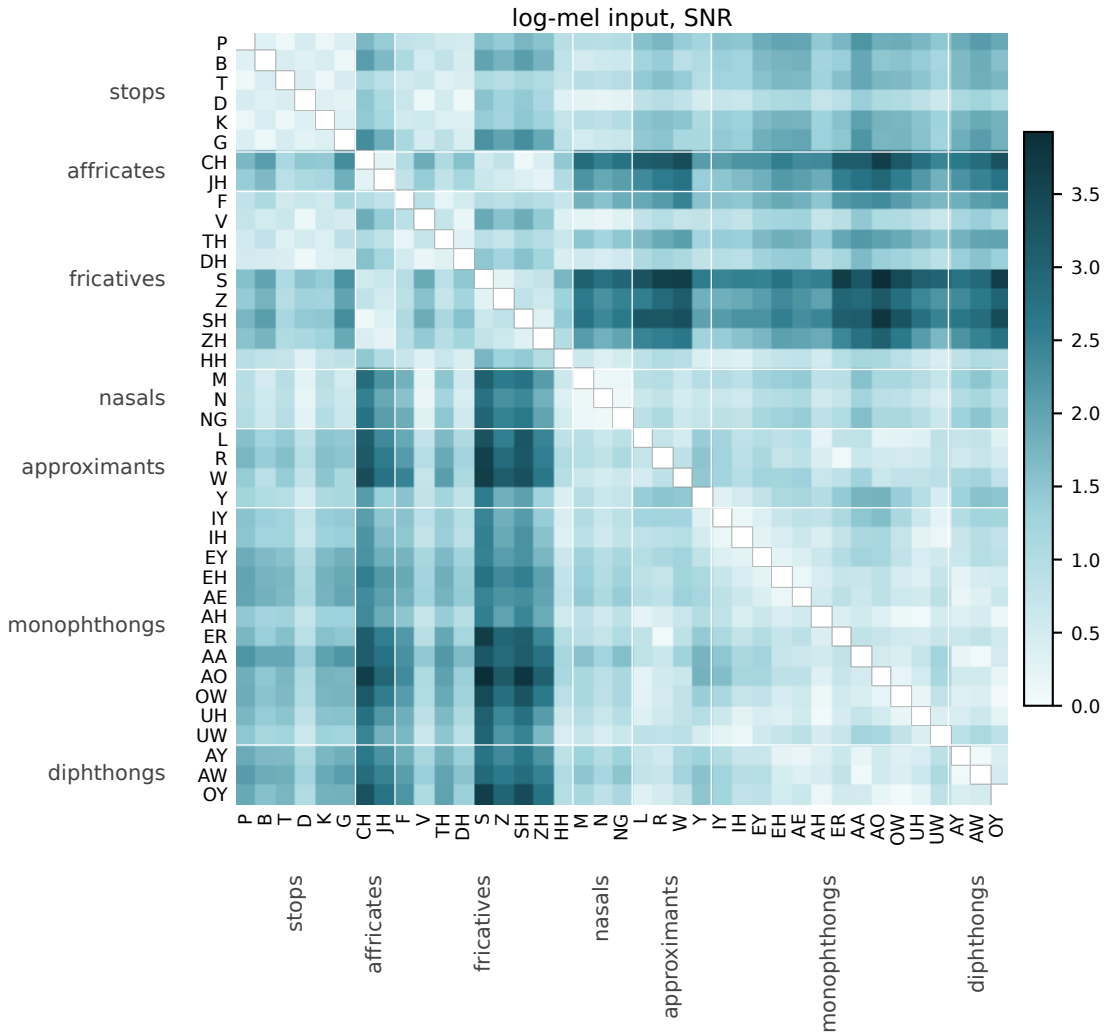


Figure B.8: Phoneme-pair separability on the log-mel input, every phoneme labeled. The matrix of Figure 3.11a: the pair $\mathrm{SNR}_{m=5}$, mean over pairs 1.31, dark is high, phonemes in the order of Section 3.10.5. The sibilant fricatives and the affricates are separable from every other phoneme; the stop, nasal and vowel blocks are not separated internally.

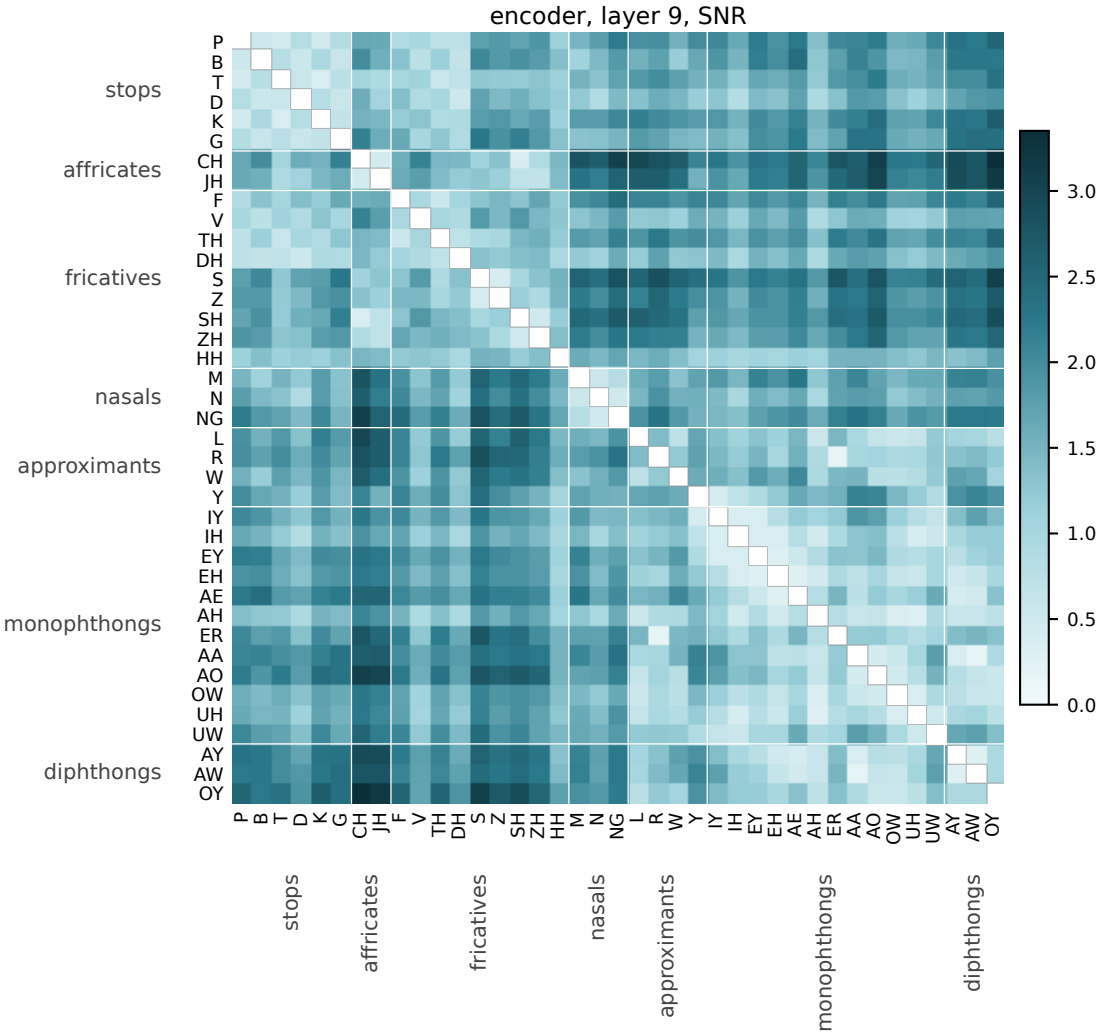


Figure B.9: Phoneme-pair separability at encoder layer 9, every phoneme labeled. The matrix of Figure 3.11b: the pair $\mathrm{SNR}_{m=5}$, mean over pairs 1.57, dark is high, phonemes in the order of Section 3.10.5.

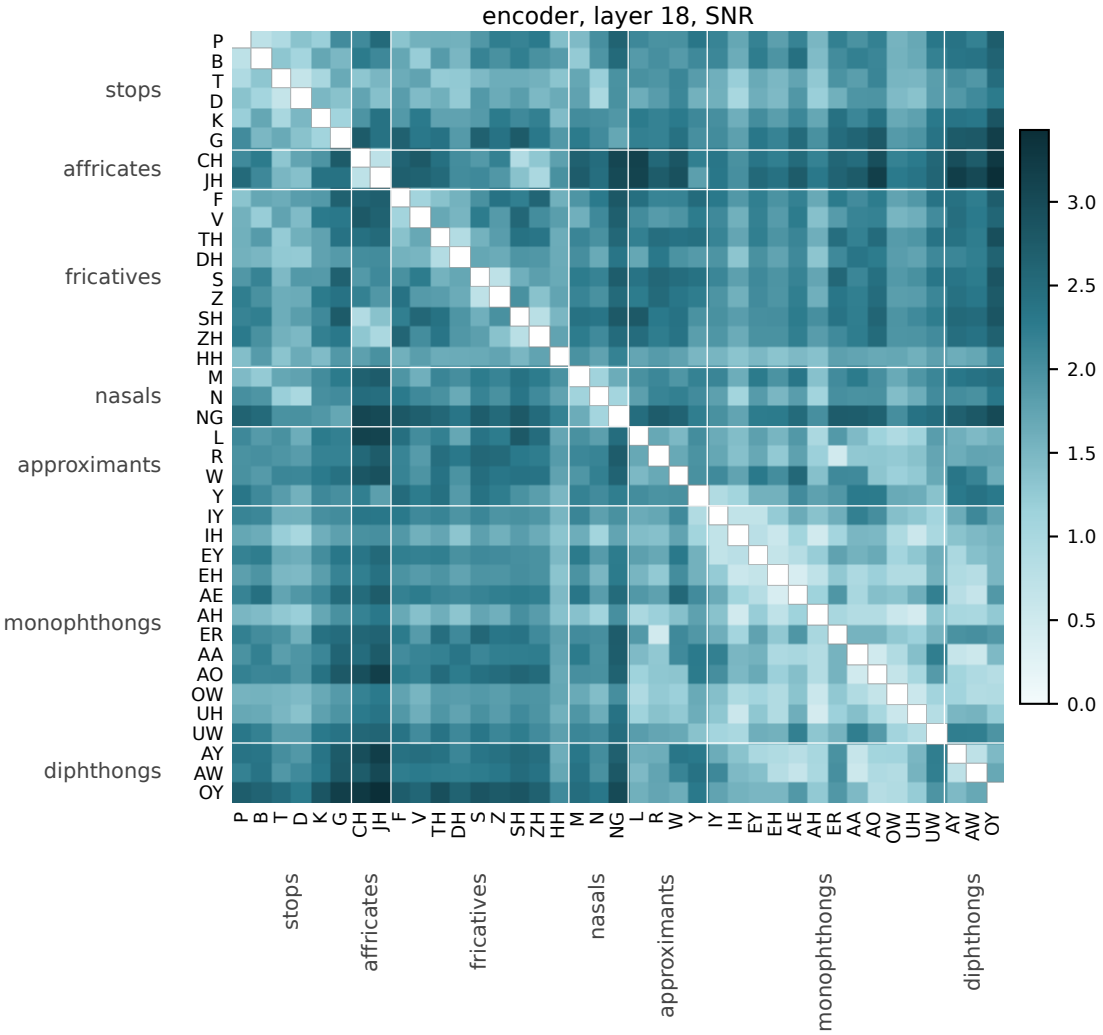


Figure B.10: Phoneme-pair separability at encoder layer 18, every phoneme labeled. The matrix of Figure 3.11c: the pair $\mathrm{SNR}_{m=5}$, mean over pairs 1.89, dark is high, phonemes in the order of Section 3.10.5. The lightest cells are the voicing pairs /P/–/B/, /T/–/D/, /K/–/G/, /F/–/V/, /TH/–/DH/, /S/–/Z/ and /SH/–/ZH/, the nasal pair /M/–/N/, and the neighboring vowels /IH/, /EH/, /AE/ and /AH/.

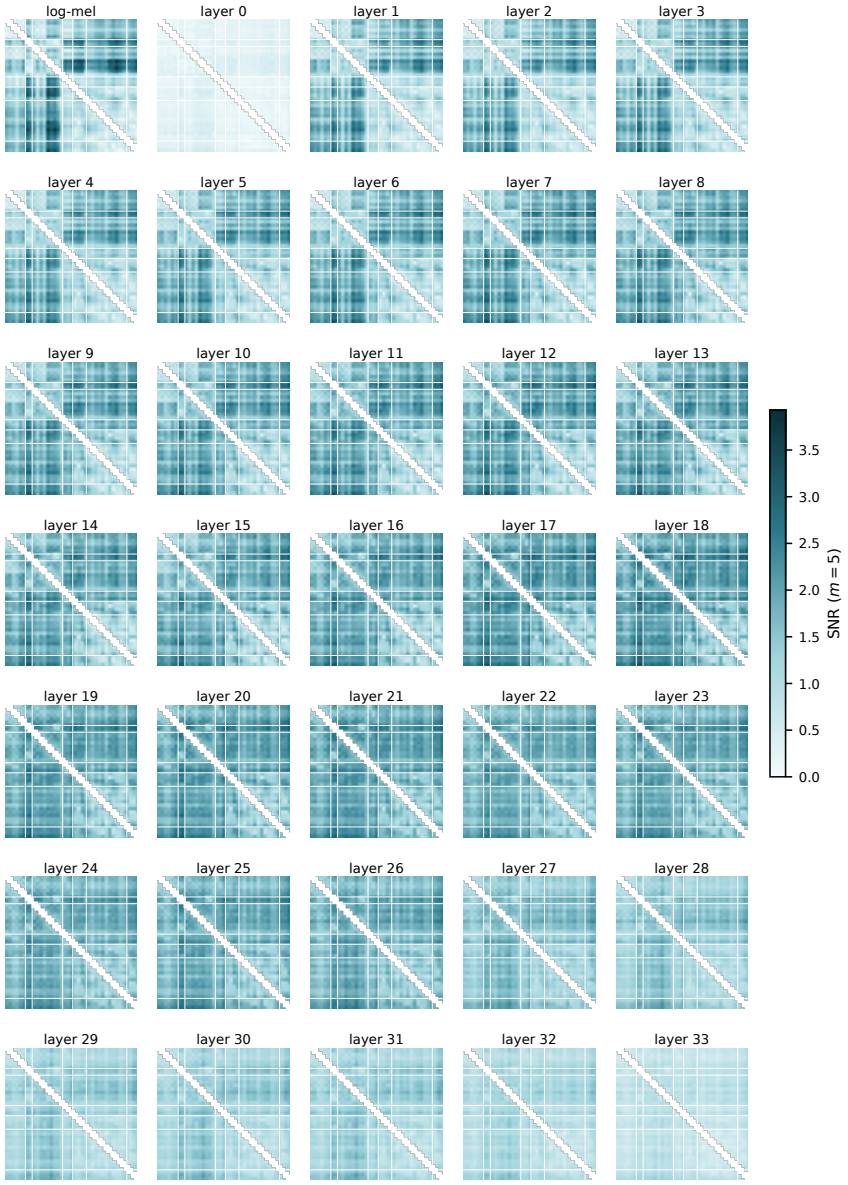


Figure B.11: Phoneme-pair separability at every stage of the encoder. Each matrix is the pair $\mathrm{SNR}_{m=5}$ of every pair of the 39 phonemes, the two directions of a pair averaged, at the log-mel input and at encoder layers 0 to 33, on one color scale from 0 to the largest pair value over all stages (3.93, on the spectrogram); dark is high. Phonemes are in the order of Section 3.10.5, with lines between manner classes; the labeled versions of the input, layer 9 and layer 18 matrices are Figures B.8–B.10. Layer 0 is the output of the convolutional stem, where the positional embedding dominates the pooled vector (Section 3.10.5).

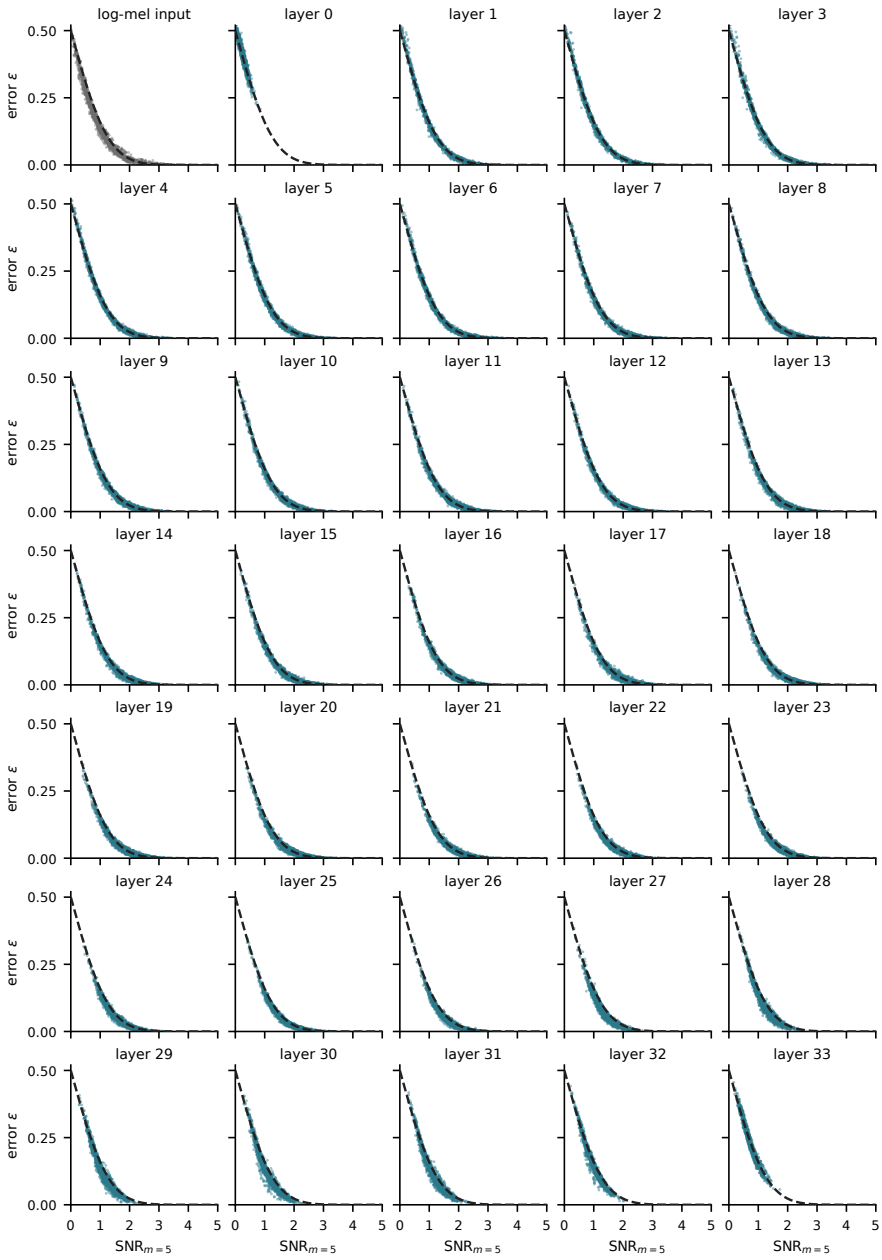


Figure B.12: Measured error of the five-shot prototype readout against the signal-to-noise ratio $\mathrm{SNR}_{m=5}$ of Eq. B.1 for all 39×38 ordered phoneme pairs, at the log-mel input (grey) and at every encoder layer 0 to 33 (teal); the black dashed curve is the prediction $\varepsilon = H(\mathrm{SNR})$ of Eq. 2.8.

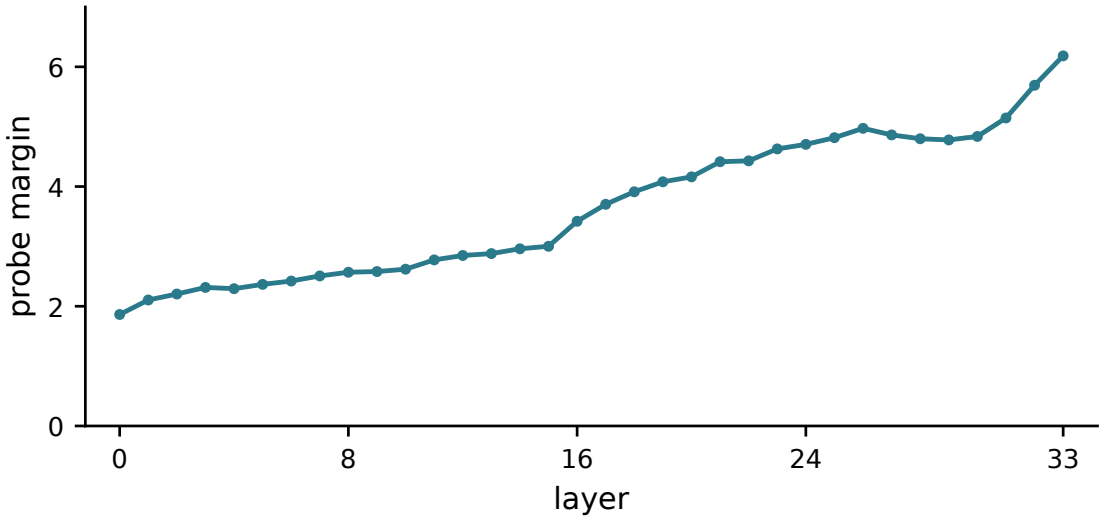


Figure B.13: Margin of the 39-way linear probe on the phoneme manifolds against encoder layer: the mean over correctly classified held-out occurrences, of the difference between the logit of the true class and the largest other logit, under five-fold cross-validation over occurrences. The line is the mean and the band $\pm$ one s.e.m. across the five folds.

B.5.5 The bias–variance and noise–noise terms

Section 3.7 follows the signal, the dimensionality and the signal–noise overlap term of the phoneme manifolds. Figure B.14 shows the two remaining noise terms of Eq. B.1. From layer 1 onward the bias–variance term lies between 0.0014 and 0.0022 and the noise–noise overlap term between 0.0019 and 0.0071; their shares of the sum of the four noise terms are 0.05 to 0.08 and 0.14 to 0.21 at layers 1, 18 and 33. The bias term is below 0.016 at every layer and below 4% of the numerator $\|\Delta \mathbf{x}_0\|^2$ + bias from layer 1 onward.

B.5.6 Word organization within phoneme manifolds

Figure B.15 shows the measure cited in Section 3.7. For each phoneme manifold and each layer, the mean squared distance between two occurrences of the phoneme drawn from the same word is divided by the mean squared distance between two occurrences drawn from different words; host

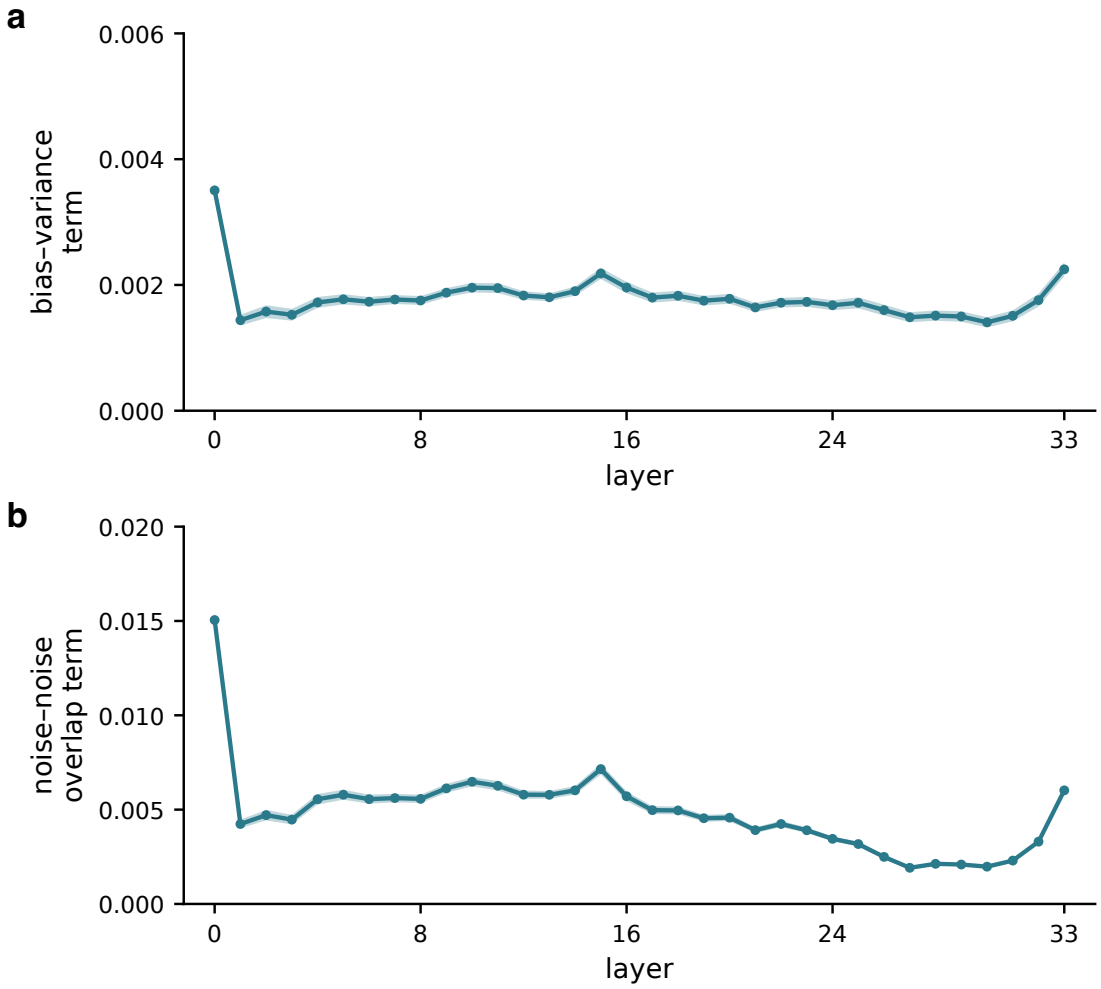


Figure B.14: The two noise terms of Eq. B.1 not followed in Section 3.7, for the phoneme manifolds at $m = 5$, against encoder layer (teal). (a) The bias–variance term β_{ab}/m^2. (b) The noise–noise overlap term. Lines are means over all 39×38 ordered pairs and bands $\pm$ one s.e.m. across the 39 phoneme classes.

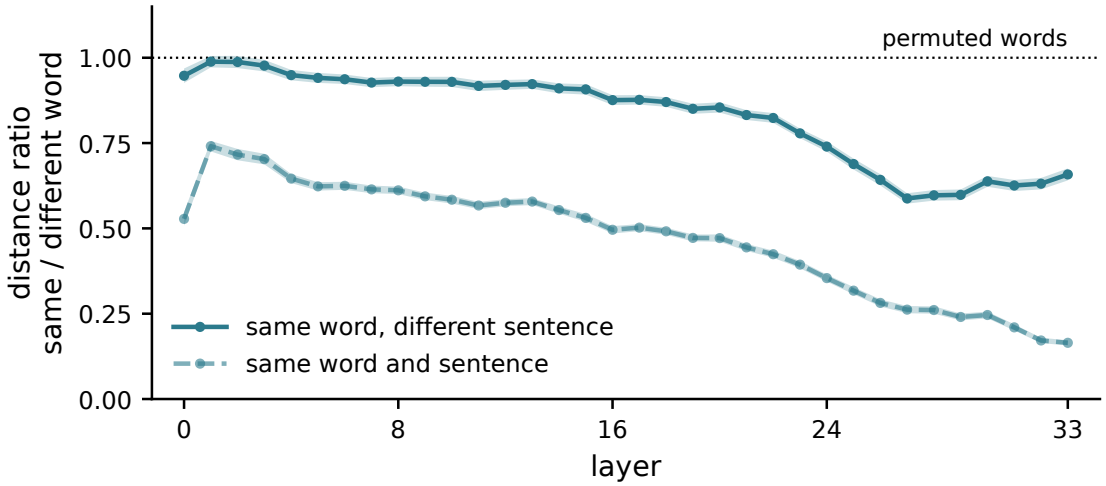


Figure B.15: Word organization within phoneme manifolds across the Whisper encoder. For each phoneme manifold, the mean squared distance between two occurrences of the phoneme drawn from the same word, divided by the mean squared distance between two occurrences drawn from different words, across encoder layers. Solid line: same word in different TIMIT sentences (15,256 pairs over the 39 phonemes). Dashed line: same word in the same sentence, read by different speakers (22,819 pairs over 38 phonemes; /ZH/ has too few). The dotted line at 1 is the ratio after the word labels are permuted within each manifold. Lines are the mean and bands ± one s.e.m. across phoneme classes.

words are the TIMIT word transcriptions of the selected occurrences. Two variants are shown: same word in different sentences, which controls for a shared recording, and same word in the same TIMIT sentence read by different speakers, which adds a shared sentence context. Occurrences drawn from the same word in different sentences are as far apart as occurrences from different words at layer 1, a ratio of 0.99; the ratio decreases to 0.83 at layer 21 and 0.59 at layer 27, and is 0.66 at the encoder output. With the sentence also shared, the ratio is 0.74 at layer 1 and 0.16 at the output. The ratio after permuting the word labels within each manifold lies between 0.99 and 1.00 at every layer.

B.6 Supplementary analyses of sentence context

B.6.1 Stimulus inclusion rule

Section 3.8 compares the input conditions over all 160,000 occurrences, without regard to whether Whisper transcribes them correctly. The control reported here compares the in-sentence condition with the isolated word at the first frame, the two conditions for which the transcription pass was run; the position-matched control of Appendix B.6.2 was not transcribed. The alternative is to restrict the comparison to occurrences transcribed correctly in both conditions, so that every manifold point is one the model has, in the operational sense, recognized. That rule is too strict to be primary: being transcribed correctly in context is itself one of the effects under study, and because accuracy is lower in isolation (Section 3.10.4) the subset is selected mainly by the single-word outcome, favoring the acoustically clearest occurrences of each word and so shrinking every manifold in a way that is not neutral between conditions. It is nonetheless a useful control, since it rules out the possibility that the decoder's in-sentence advantage is carried by occurrences the model fails on in isolation. Here the analysis of Figure 3.12 is repeated on the restricted subset of 118,041 occurrences; 797 of the 800 words retain at least $m+1 = 6$ occurrences, so that the readout is defined, and the three that do not are dropped from the restricted comparison, leaving 118,039 occurrences. Because the words then differ in their number of occurrences, each word's covariance is normalized by its own count, and the estimator of Appendix B.3 is otherwise unchanged; the error is the predicted $H(\mathrm{SNR}_{m=5})$ rather than the measured readout.

The restriction favors the isolated word, as expected, and leaves every pattern of Section 3.8 in place. In the encoder the single-word ratio rises from 5.75 to 6.21 at layer 27 and from 4.54 to 4.88 at layer 33, whereas the in-sentence ratio is nearly unchanged, 5.29 and 1.53 at the same layers against 5.19 and 1.54 on all occurrences; the in-sentence ratio remains below the single-word ratio at every encoder layer, and the collapse at the encoder output is the same, to 0.31 of the single-word

value with $D = 6.6$. In the decoder, the crossover remains at layer 46, at 0.93 in both conditions, and the in-sentence ratio is again the higher of the two over layers 47 to 55. At the peak, the restriction removes the difference between conditions: the single-word ratio rises from 6.01 to 7.01 at layer 63 and the in-sentence ratio from 6.61 to 7.03, so the two coincide, while the contrast that produces them is unchanged, with signal 2.13 against 3.94 and D of 36.6 against 14.9. Over layers 56 to 59, the in-sentence ratio falls slightly below the single-word ratio on the restricted subset. After the peak the divergence of the full comparison is preserved: at the decoder output the ratio is 6.01 in context against 4.89 in isolation, and the predicted error 2.6×10^{-5} against 2.7×10^{-4}. The decoder's in-sentence separability therefore does not come from occurrences the model fails on in isolation; what the restriction changes is the height of the single-word peak, which is why Section 3.8 does not base its claim on comparing the two peaks.

B.6.2 Position-matched control

Section 3.8 compares the in-sentence condition with the position-matched control rather than with the isolated word of the preceding sections, because the isolated word differs from the in-sentence word in its position within the input window as well as in the speech that precedes it: it begins on the first frame, while the in-sentence word begins on the frame at which it falls in its sentence, so the two receive different sinusoidal positional embeddings in the encoder. This appendix reports the three conditions together. The position-matched control of Section 3.10.4 removes the positional difference. The isolated word is preceded by silence up to its in-sentence onset frame, and the decoder input is the prefix of the single-word condition. In the encoder, the control therefore differs from the in-sentence condition only in the preceding audio; in the decoder, it differs in the preceding audio and in the decoder input, since the preceding tokens set the in-sentence readout positions. Figure B.16a shows the three conditions, and Figure B.16b shows the signal-to-noise ratio of the control relative to the two main conditions.

In the encoder, the control lies between the two conditions through the middle of the stack, at 1.64, 2.82 and 5.00 at layers 9, 19 and 27 against 1.35, 2.39 and 5.19 in context and 2.28, 3.73 and 5.75 in isolation. Position thus accounts for about two-thirds of the in-sentence deficit at layers 9 and 19; the preceding speech costs between 0.2 and 0.6 in the ratio over layers 6 to 26, less than 0.1 over layers 1 to 5, and less than 0.2 from the peak through layer 30, where the control and the in-sentence condition coincide. At layer 0, the control gives the same 0.08 as the in-sentence condition, against 0.93 in isolation, since the within-word variability at the output of the stem is the positional embedding. The control also reproduces the in-sentence radius in part: R_a^2 is 5.8 at layer 9 and 5.7 at layer 27, against 7.7 and 7.3 in context and 4.0 and 4.2 in isolation, with the same or a smaller D than the isolated word. Over the last encoder layers, the control does not collapse. At layer 33 its ratio is 3.78, against 4.54 in isolation and 1.54 in context, its squared radius 52 against 32 and 126, and its D 12.7 against 13.1 and 7.1; the position costs the same 0.8 in the ratio here as through the middle of the stack, and the remaining 2.2 of the in-sentence deficit at the encoder output is the preceding speech. The prototype error of the control at layer 33 is 2.2×10^{-3}, against 1.4×10^{-3} in isolation and 0.070 in context.

In the decoder, the control follows the isolated word. At layer 34 its ratio is 0.74 against 0.78, with squared radius 0.46 against 0.76 and D of 11.4 against 9.7, where the in-sentence condition gives 0.50, 5.2 and 21.9; the two remain within 0.3 of one another through layer 46, and at the decoder output the control gives 4.44 against 4.39, with D of 13.6 against 10.2, where the in-sentence ratio is 5.79 with $D = 28.6$. The high-dimensional within-word variability of the in-sentence decoder and the divergence after the peak are therefore produced by the preceding tokens, not by the word's position in the audio. Over layers 47 to 65, the control lies below the isolated word, reaching 5.24 at layer 63, within 0.01 of its maximum at layer 64, against 6.01; the both-correct control of Appendix B.6.1 was not run for this condition, and Section 3.8 does not conclude on the height of this peak. The control's prototype error in the decoder is 0.22 at layer 34 against 0.21 in isolation

and 0.33 in context, and 9.6×10^{-4} at layer 63 against 3.2×10^{-4} and 5.7×10^{-4}.

B.6.3 Radius and centroid separation

The signal of Figure 3.13a is the squared centroid separation of a word pair divided by the squared radius of the reference word's manifold. Figure B.17 shows the two parts separately for both conditions. In the encoder, the squared radius is larger in context at every layer, 7.7 against 4.0 at layer 9 and 7.3 against 4.2 at layer 27, whereas the squared centroid separation is unchanged or larger, 3.9 against 3.7 at layer 9 and 15.4 against 12.2 at layer 27; the lower in-sentence signal is therefore produced by larger manifolds, not by centroids that lie closer together. At layer 0, the in-sentence radius is 234 against 17, the positional embedding of the word's frames. Over the last encoder layers, it grows from 7.3 at layer 27 to 126 at layer 33 against 32 in isolation, while the separation grows from 15.4 to 104 against 84. In the decoder, both parts change: at layer 34 the in-sentence radius is 5.2 against 0.76, and at the peak, layer 63, the radius is 82 against 61 and the separation 159 against 199, so the smaller in-sentence signal at the peak combines a larger radius with a smaller separation. Both quantities rise steeply at the final layer normalization, layer 66, in both conditions.

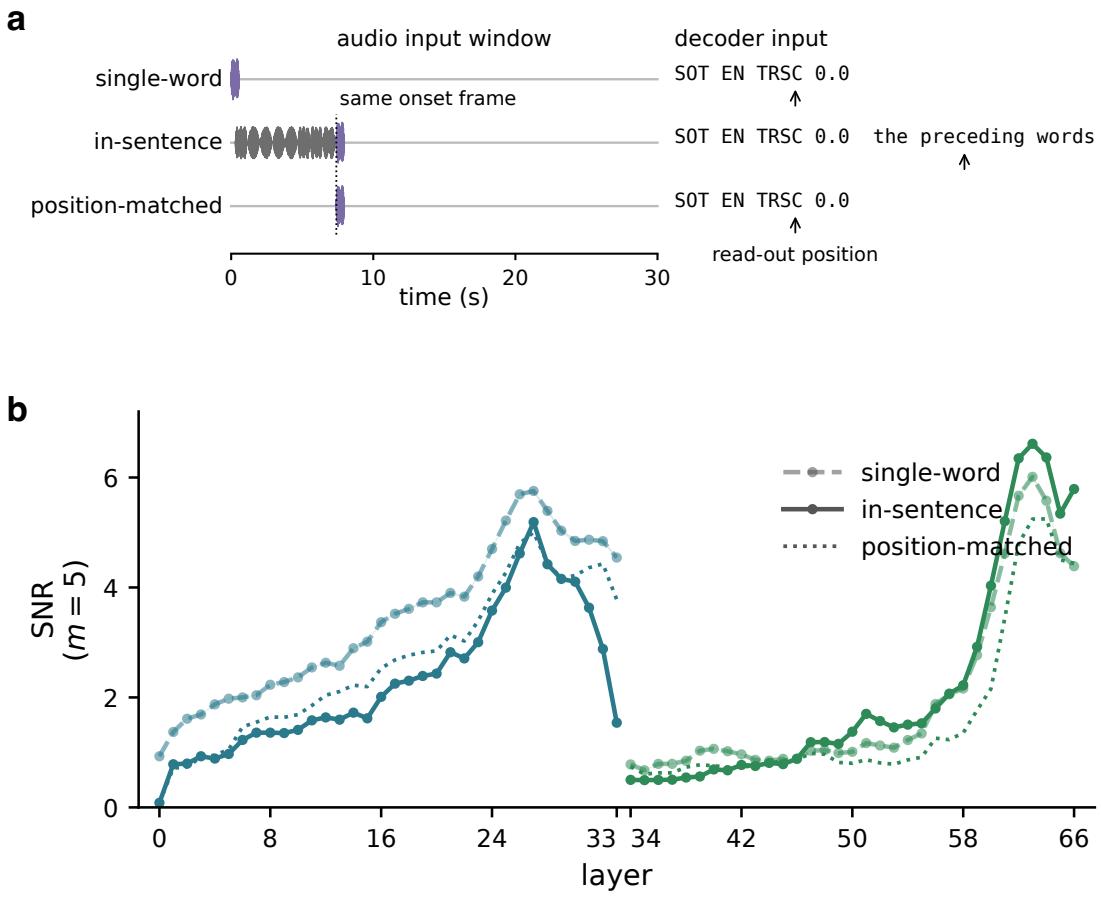


Figure B.16: The three input conditions and the position-matched control, and the effect of sentence context. (a) The three conditions. Each row is one 30 s input window; the target word's waveform (purple) is the same in every row. Single-word: the word at the first frame, zeros after it. In-sentence: the sentence audio (grey) from its onset through the word, zeros after it. Position-matched: silence up to the word's in-sentence onset, then the word. To the right, the decoder input for each condition: the task prefix of Figure 3.1 followed in the in-sentence condition by the tokens of the preceding words, with the readout position marked. Waveforms are schematic. (b) Mean $\mathrm{SNR}_{m=5}$ of Eq. B.1 over all 800×799 ordered pairs at every layer for the single-word condition (dashed), the in-sentence condition (solid) and the position-matched control (dotted); encoder in teal (layers 0–33), decoder in green (layers 34–66). In the encoder, the control lies between the two conditions through the middle of the stack, coincides with the in-sentence condition at the peak, 5.00 against 5.19 at layer 27, and stays near the isolated word at the encoder output, 3.78 against 4.54, where the in-sentence ratio is 1.54. In the decoder, the control follows the isolated word through layer 46 (0.74 vs 0.78 at layer 34), lies below it from layers 47 to 65, and rejoins it at the output (4.44 vs 4.39). Bands on the two main conditions are $\pm$ one s.e.m. across the 800 words and are smaller than the markers.

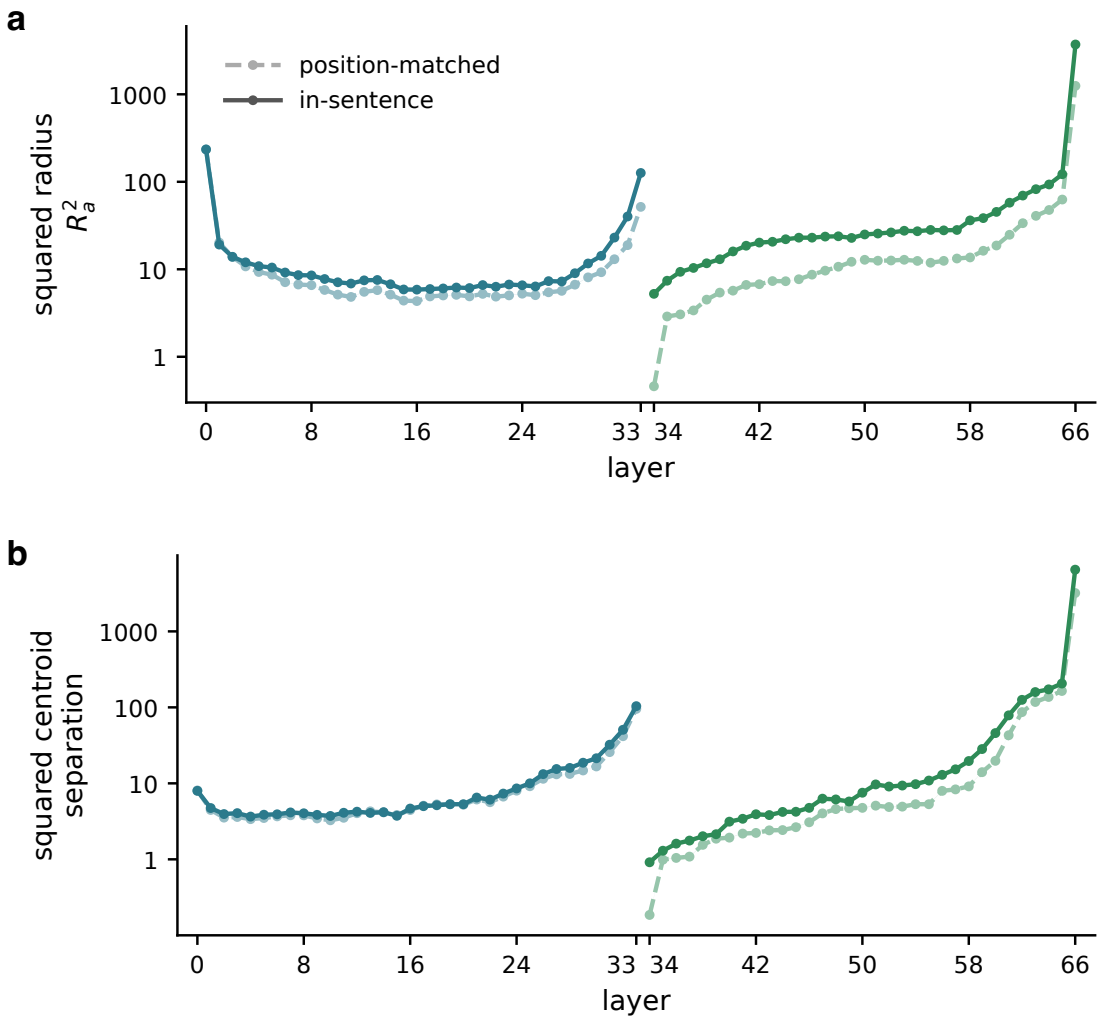


Figure B.17: The two parts of the signal with and without sentence context. (a) Squared manifold radius R_a^2, mean over the 800 words, and (b) squared centroid separation, mean over all 800×799 ordered pairs, at every layer for the single-word condition (dashed) and the in-sentence condition (solid); encoder in teal (layers 0–33), decoder in green (layers 34–66); logarithmic axes. Bands are $\pm$ one s.e.m. across the 800 words.

References

[1] Alvin M. Liberman, Franklin S. Cooper, Donald P. Shankweiler, and Michael Studdert-Kennedy. Perception of the speech code. *Psychological Review*, 74(6):431–461, 1967.

[2] Laura Gwilliams, Jean-Rémi King, Alec Marantz, and David Poeppel. Neural dynamics of phoneme sequences reveal position-invariant code for content and order. *Nature Communications*, 13(1):6606, 2022.

[3] Patricia A Carpenter, Marcel A Just, and Peter Shell. What one intelligence test measures: a theoretical account of the processing in the Raven Progressive Matrices Test. *Psychological Review*, 97(3):404–431, 1990.

[4] Mattia Rigotti, Omri Barak, Melissa R Warden, Xiao-Jing Wang, Nathaniel D Daw, Earl K Miller, and Stefano Fusi. The importance of mixed selectivity in complex cognitive tasks. *Nature*, 497(7451):585–590, 2013.

[5] James J DiCarlo and David D Cox. Untangling invariant object recognition. *Trends in Cognitive Sciences*, 11(8):333–341, 2007.

[6] James J DiCarlo, Davide Zoccolan, and Nicole C Rust. How does the brain solve visual object recognition? *Neuron*, 73(3):415–434, 2012.

[7] Chou P Hung, Gabriel Kreiman, Tomaso Poggio, and James J DiCarlo. Fast readout of object identity from macaque inferior temporal cortex. *Science*, 310(5749):863–866, 2005.

[8] Nima Mesgarani, Connie Cheung, Keith Johnson, and Edward F. Chang. Phonetic feature encoding in human superior temporal gyrus. *Science*, 343(6174):1006–1010, 2014. doi: 10.1126/science.1245994.

[9] Han Gyol Yi, Matthew K. Leonard, and Edward F. Chang. The encoding of speech sounds in the superior temporal gyrus. *Neuron*, 102(6):1096–1110, 2019. doi: 10.1016/j.neuron.2019.04.023.

[10] Iain DeWitt and Josef P. Rauschecker. Phoneme and word recognition in the auditory ventral stream. *Proceedings of the National Academy of Sciences*, 109(8):E505–E514, 2012. doi: 10.1073/pnas.1113427109.

[11] David J Freedman, Maximilian Riesenhuber, Tomaso Poggio, and Earl K Miller. A comparison of primate prefrontal and inferior temporal cortices during visual categorization. *Journal of Neuroscience*, 23(12):5235–5246, 2003.

[12] Silvia Bernardi, Marcus K Benna, Mattia Rigotti, Jérôme Munuera, Stefano Fusi, and C Daniel Salzman. The geometry of abstraction in the hippocampus and prefrontal cortex. *Cell*, 183(4):954–967, 2020.

[13] SueYeon Chung, Daniel D Lee, and Haim Sompolinsky. Classification and geometry of general perceptual manifolds. *Physical Review X*, 8(3):031003, 2018.

[14] Uri Cohen, SueYeon Chung, Daniel D Lee, and Haim Sompolinsky. Separability and geometry of object manifolds in deep neural networks. *Nature communications*, 11(1):746, 2020.

[15] Ben Sorscher, Surya Ganguli, and Haim Sompolinsky. Neural representational geometry underlies few-shot concept learning. *Proceedings of the National Academy of Sciences*, 119 (43):e2200800119, 2022.

[16] Daniel L. K. Yamins, Ha Hong, Charles F. Cadieu, Ethan A. Solomon, Darren Seibert, and James J. DiCarlo. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences*, 111(23):8619–8624, 2014. doi: 10.1073/pnas.1403112111.

[17] Alexander J. E. Kell, Daniel L. K. Yamins, Erica N. Shook, Sam V. Norman-Haignere, and Josh H. McDermott. A task-optimized neural network replicates human auditory behavior, predicts brain responses, and reveals a cortical processing hierarchy. *Neuron*, 98(3):630–644.e16, 2018. doi: 10.1016/j.neuron.2018.03.044.

[18] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *International Conference on Machine Learning*, pages 28492–28518. PMLR, 2023.

[19] Daniel C Krawczyk. The cognition and neuroscience of relational reasoning. *Brain Research*, 1428:13–23, 2012. doi: 10.1016/j.brainres.2010.11.080.

[20] François Chollet. On the measure of intelligence. *arXiv preprint arXiv:1911.01547*, 2019.

[21] James C Raven. Mental tests used in genetic studies: The performance of related individuals on tests mainly educative and mainly reproductive. Msc thesis, University of London, 1936.

[22] Linda BL Vodegel Matzen, Maurits W Van der Molen, and Ad CM Dudink. Error analysis of Raven test performance. *Personality and Individual Differences*, 16(3):433–445, 1994.

[23] Laura E Matzen, Zachary O Benz, Kevin R Dixon, Jamie Posey, James K Kroger, and Ann E Speed. Recreating Raven's: Software for systematically generating large numbers of Raven-like matrix problems with normed properties. *Behavior Research Methods*, 42(2):525–541, 2010. doi: 10.3758/BRM.42.2.525.

[24] Matthew V Chafee, Bruno B Averbeck, and David A Crowe. Representing spatial relationships in posterior parietal cortex: single neurons code object-referenced position. *Cerebral Cortex*, 17(12):2914–2932, 2007.

[25] Yukako Yamane, Kazushige Tsunoda, Madoka Matsumoto, Adam N Phillips, and Manabu Tanifuji. Representation of the spatial relationship among object parts by neurons in macaque inferotemporal cortex. *Journal of neurophysiology*, 96(6):3147–3156, 2006.

[26] Tyler Davis, Micah Goldwater, and Josue Giron. From concrete examples to abstract relations: The rostrolateral prefrontal cortex integrates novel examples into relational categories. *Cerebral Cortex*, 27(4):2652–2670, 2017.

[27] Hristos S Courellis, Juri Minxha, Araceli R Cardenas, Daniel L Kimmel, Chrystal M Reed, Taufik A Valiante, C Daniel Salzman, Adam N Mamelak, Stefano Fusi, and Ueli Rutishauser. Abstract representations emerge in human hippocampal neurons during inference. *Nature*, 632(8026):841–849, 2024.

[28] Christopher Summerfield, Fabrice Luyckx, and Hannah Sheahan. Structure learning and the posterior parietal cortex. *Progress in Neurobiology*, 184:101717, 2020.

[29] Stephanie Nelli, Lukas Braun, Tsvetomira Dumbalska, Andrew Saxe, and Christopher Summerfield. Neural knowledge assembly in humans and neural networks. *Neuron*, 111(9): 1504–1516.e9, 2023. doi: 10.1016/j.neuron.2023.02.014.

[30] David Barrett, Felix Hill, Adam Santoro, Ari Morcos, and Timothy Lillicrap. Measuring abstract reasoning in neural networks. In *International Conference on Machine Learning*, pages 511–520. PMLR, 2018.

[31] Chi Zhang, Feng Gao, Baoxiong Jia, Yixin Zhu, and Song-Chun Zhu. RAVEN: A dataset for relational and analogical visual reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5312–5322, 2019. doi: 10.1109/CVPR.2019.00546.

[32] Sheng Hu, Yuqing Ma, Xianglong Liu, Yanlu Wei, and Shihao Bai. Stratified rule-aware network for abstract visual reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1567–1574, 2021. doi: 10.1609/aaai.v35i2.16248.

[33] Yuhuai Wu, Honghua Dong, Roger Grosse, and Jimmy Ba. The scattering compositional learner: Discovering objects, attributes, relationships in analogical reasoning. *arXiv preprint arXiv:2007.04212*, 2020.

[34] Xander Steenbrugge, Sam Leroux, Tim Verbelen, and Bart Dhoedt. Improving generalization for abstract reasoning tasks using disentangled feature representations. *arXiv preprint arXiv:1811.04784*, 2018.

[35] Michael Hersche, Mustafa Zeqiri, Luca Benini, Abu Sebastian, and Abbas Rahimi. A neuro-vector-symbolic architecture for solving Raven's progressive matrices. *Nature Machine Intelligence*, 5(4):363–375, 2023.

[36] Nasim Rahaman, Muhammad Waleed Gondal, Shruti Joshi, Peter Gehler, Yoshua Bengio, Francesco Locatello, and Bernhard Schölkopf. Dynamic inference with neural interpreters. *Advances in Neural Information Processing Systems*, 34:10985–10998, 2021.

[37] Chi Zhang, Sirui Xie, Baoxiong Jia, Ying Nian Wu, Song-Chun Zhu, and Yixin Zhu. Learning algebraic representation for systematic generalization in abstract reasoning. In *European Conference on Computer Vision*, pages 692–709. Springer, 2022.

[38] Chi Zhang, Baoxiong Jia, Feng Gao, Yixin Zhu, Hongjing Lu, and Song-Chun Zhu. Learning perceptual inference by contrasting. *Advances in Neural Information Processing Systems*, 32, 2019.

[39] Shanka Subhra Mondal, Jonathan D Cohen, and Taylor W Webb. Slot abstractors: Toward scalable abstract visual reasoning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 36088–36105, 2024.

[40] Mikołaj Małkiński and Jacek Mańdziuk. Deep learning methods for abstract visual reasoning: A survey on Raven's progressive matrices. *arXiv preprint arXiv:2201.12382*, 2022.

[41] Yizhe Zhang, He Bai, Ruixiang Zhang, Jiatao Gu, Shuangfei Zhai, Josh Susskind, and Navdeep Jaitly. How far are we from intelligent visual deductive reasoning? In *First Conference on Language Modeling (COLM)*, 2024.

[42] Pu Jian, Junhong Wu, Wei Sun, Chen Wang, Shuo Ren, and Jiajun Zhang. Look again, think slowly: Enhancing visual reflection in vision-language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 9262–9281, 2025.

[43] Kanzhi Cheng, Yantao Li, Fangzhi Xu, Jianbing Zhang, Hao Zhou, and Yang Liu. Vision-language models can self-improve reasoning via reflection. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8876–8892, 2025. doi: 10.18653/v1/2025.naacl-long.447.

[44] Chenyu Lin, Cheng Chi, Jinlin Wu, Sharon Li, and Kaiyang Zhou. Learning to think fast and slow for visual language models. *arXiv preprint arXiv:2511.16670*, 2025.

[45] Haozhe Wang, Chao Qu, Zuming Huang, Wei Chu, Fangzhen Lin, and Wenhu Chen. VL-Rethinker: Incentivizing self-reflection of vision-language models with reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 38, pages 30865–30891, 2025.

[46] Chao Chen, Zhixin Ma, Yongqi Li, Yupeng Hu, Yinwei Wei, Wenjie Li, and Liqiang Nie. Reasoning in the dark: Interleaved vision-text reasoning in latent space. In *Findings of the Association for Computational Linguistics: ACL 2026*, pages 39117–39129, 2026. doi: 10.18653/v1/2026.findings-acl.1949.

[47] Thomas Miconi and Kenneth Kay. Neural mechanisms of relational learning and fast knowledge reassembly in plastic neural networks. *Nature Neuroscience*, 28(2):406–414, 2025. doi: 10.1038/s41593-024-01852-8.

[48] Samuel Lippl, Kenneth Kay, Greg Jensen, Vincent P Ferrera, and LF Abbott. A mathematical theory of relational generalization in transitive inference. *Proceedings of the National Academy of Sciences*, 121(28):e2314511121, 2024.

[49] Kenneth Kay, Natalie Biderman, Ramin Khajeh, Manuel Beiran, Christopher J Cueva, Daphna Shohamy, Greg Jensen, Xue-Xin Wei, Vincent P Ferrera, and LF Abbott. Emergent neural dynamics and geometry for generalization in a transitive inference task. *PLOS Computational Biology*, 20(4):e1011954, 2024.

[50] François Fleuret, Ting Li, Charles Dubout, Emma K Wampler, Steven Yantis, and Donald Geman. Comparing machines and humans on a visual categorization test. *Proceedings of the National Academy of Sciences*, 108(43):17621–17625, 2011.

[51] Junkyung Kim, Matthew Ricci, and Thomas Serre. Not-So-CLEVR: learning same–different relations strains feedforward neural networks. *Interface Focus*, 8(4):20180011, 2018.

[52] Aimen Zerroug, Mohit Vaishnav, Julien Colin, Sebastian Musslick, and Thomas Serre. A benchmark for compositional visual reasoning. *Advances in Neural Information Processing Systems*, 35:29776–29788, 2022.

[53] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[54] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

[55] Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. *Advances in Neural Information Processing Systems*, 33:11525–11538, 2020.

[56] Taylor W Webb, Ishan Sinha, and Jonathan D Cohen. Emergent symbols through binding in external memory. In *International Conference on Learning Representations*, 2021.

[57] Taylor W Webb, Steven M Frankland, Awni Altabaa, Simon Segert, Kamesh Krishnamurthy, Declan Campbell, Jacob Russin, Tyler Giallanza, Randall O'Reilly, John Lafferty, and Jonathan D Cohen. The relational bottleneck as an inductive bias for efficient abstraction. *Trends in Cognitive Sciences*, 28(9):829–843, 2024. doi: 10.1016/j.tics.2024.04.001.

[58] Alex Graves, Greg Wayne, and Ivo Danihelka. Neural Turing machines. *arXiv preprint arXiv:1410.5401*, 2014.

[59] Qingyang Wu, Zhenzhong Lan, Kun Qian, Jing Gu, Alborz Geramifard, and Zhou Yu. Memformer: A memory-augmented transformer for sequence modeling. In *Findings of the Association for Computational Linguistics: AACL-IJCNLP 2022*, pages 308–318, 2022.

[60] Aydar Bulatov, Yury Kuratov, and Mikhail Burtsev. Recurrent memory transformer. *Advances in Neural Information Processing Systems*, 35:11079–11091, 2022.

[61] Yikun Wang, Siyin Wang, Qinyuan Cheng, Zhaoye Fei, Liang Ding, Qipeng Guo, Dacheng Tao, and Xipeng Qiu. VisuoThink: Empowering LVLM reasoning with multimodal tree search. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 21707–21719, 2025. doi: 10.18653/v1/2025.acl-long.1053.

[62] DiJia Su, Sainbayar Sukhbaatar, Michael Rabbat, Yuandong Tian, and Qinqing Zheng. Dualformer: Controllable fast and slow thinking by learning with randomized reasoning traces. In *International Conference on Learning Representations*, 2025.

[63] Lucas Lehnert, Sainbayar Sukhbaatar, DiJia Su, Qinqing Zheng, Paul Mcvay, Michael Rabbat, and Yuandong Tian. Beyond A*: Better planning with transformers via search dynamics bootstrapping. In *First Conference on Language Modeling (COLM)*, 2024.

[64] Huajie Tan, Yuheng Ji, Xiaoshuai Hao, Xiansheng Chen, Pengwei Wang, Zhongyuan Wang, and Shanghang Zhang. Reason-RFT: Reinforcement fine-tuning for visual reasoning of vision language models. In *Advances in Neural Information Processing Systems*, volume 38, pages 5772–5822, 2025.

[65] Hardy Chen, Haoqin Tu, Fali Wang, Hui Liu, Xianfeng Tang, Xinya Du, Yuyin Zhou, and Cihang Xie. SFT or RL? an early investigation into training R1-like reasoning large vision-language models. *arXiv preprint arXiv:2504.11468*, 2025.

[66] Aran Nayebi, Rishi Rajalingham, Mehrdad Jazayeri, and Guangyu Robert Yang. Neural foundations of mental simulation: Future prediction of latent representations on dynamic scenes. *Advances in Neural Information Processing Systems*, 36:70548–70561, 2023.

[67] Christopher Langdon, Mikhail Genkin, and Tatiana A Engel. A unifying perspective on neural manifolds and circuits for cognition. *Nature Reviews Neuroscience*, 24(6):363–377, 2023.

[68] John R Anderson, Michael Matessa, and Christian Lebiere. ACT-R: A theory of higher level cognition and its relation to visual attention. *Human–Computer Interaction*, 12(4):439–462, 1997.

[69] Daniel R Little, Stephan Lewandowsky, and Thomas L Griffiths. A Bayesian model of rule induction in Raven's Progressive Matrices. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 34, 2012.

[70] Peiran Gao, Eric Trautmann, Byron Yu, Gopal Santhanam, Stephen Ryu, Krishna Shenoy, and Surya Ganguli. A theory of multineuronal dimensionality, dynamics and measurement. *bioRxiv*, page 214262, 2017.

[71] Jan Te Nijenhuis, Olga F Voskuijl, and Natasja B Schijve. Practice and coaching on IQ tests: Quite a lot of g. *International Journal of Selection and Assessment*, 9(4):302–308, 2001.

[72] James R Flynn. Massive IQ gains in 14 nations: what IQ tests really measure. *Psychological Bulletin*, 101(2):171–191, 1987. doi: 10.1037/0033-2909.101.2.171.

[73] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30, 2017.

[74] Jiachen Zhu, Katrina Evtimova, Yubei Chen, Ravid Shwartz-Ziv, and Yann LeCun. Variance-covariance regularization improves representation learning. *arXiv preprint arXiv:2306.13292*, 2023.

[75] Adrien Bardes, Jean Ponce, and Yann LeCun. VICReg: Variance-invariance-covariance regularization for self-supervised learning. In *International Conference on Learning Representations*, 2022.

[76] Wenbin Wang, Yongcheng Jing, Liang Ding, Yingjie Wang, Li Shen, Yong Luo, Bo Du, and Dacheng Tao. Retrieval-augmented perception: High-resolution image perception meets visual RAG. In *International Conference on Machine Learning*, pages 63290–63307. PMLR, 2025.

[77] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

[78] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736, 2022.

[79] Gordon E. Peterson and Harold L. Barney. Control methods used in a study of the vowels. *The Journal of the Acoustical Society of America*, 24(2):175–184, 1952.

[80] Dennis H. Klatt. Linguistic uses of segmental duration in English: Acoustic and perceptual evidence. *The Journal of the Acoustical Society of America*, 59(5):1208–1221, 1976.

[81] Sven L. Mattys, Laurence White, and James F. Melhorn. Integration of multiple speech segmentation cues: A hierarchical framework. *Journal of Experimental Psychology: General*, 134(4):477–500, 2005.

[82] William D. Marslen-Wilson. Functional parallelism in spoken word-recognition. *Cognition*, 25(1–2):71–102, 1987.

[83] James L. McClelland and Jeffrey L. Elman. The TRACE model of speech perception. *Cognitive Psychology*, 18(1):1–86, 1986.

[84] George A. Miller, George A. Heise, and William Lichten. The intelligibility of speech as a function of the context of the test materials. *Journal of Experimental Psychology*, 41(5): 329–335, 1951. doi: 10.1037/h0062491.

[85] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Advances in Neural Information Processing Systems*, volume 33, pages 12449–12460. Curran Associates, Inc., 2020.

[86] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. HuBERT: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3451–3460, 2021. doi: 10.1109/TASLP.2021.3122291.

[87] Yonatan Belinkov and James Glass. Analyzing hidden representations in end-to-end automatic speech recognition systems. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[88] Ankita Pasad, Ju-Chieh Chou, and Karen Livescu. Layer-wise analysis of a self-supervised speech representation model. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 914–921, 2021. doi: 10.1109/ASRU51503.2021.9688093.

[89] Ankita Pasad, Chung-Ming Chien, Shane Settle, and Karen Livescu. What do self-supervised speech models know about words? *Transactions of the Association for Computational Linguistics*, 12:372–391, 2024. doi: 10.1162/tacl_a_00656.

[90] Cory Stephenson, Jenelle Feather, Suchismita Padhy, Oguz Elibol, Hanlin Tang, Josh McDermott, and SueYeon Chung. Untangling in invariant speech recognition. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[91] Kawin Ethayarajh. How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65, 2019. doi: 10.18653/v1/D19-1006.

[92] Jonathan Mamou, Hang Le, Miguel Del Rio, Cory Stephenson, Hanlin Tang, Yoon Kim, and SueYeon Chung. Emergence of separable manifolds in deep language representations. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 6713–6723, 2020.

[93] Juliette Millet, Charlotte Caucheteux, Pierre Orhan, Yves Boubenec, Alexandre Gramfort, Ewan Dunbar, Christophe Pallier, and Jean-Rémi King. Toward a realistic model of speech processing in the brain with self-supervised learning. In *Advances in Neural Information Processing Systems*, volume 35, pages 33428–33443. Curran Associates, Inc., 2022.

[94] Aditya R. Vaidya, Shailee Jain, and Alexander G. Huth. Self-supervised models of audio effectively explain human cortical responses to speech. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 21927–21944. PMLR, 2022.

[95] Yuanning Li, Gopala K. Anumanchipalli, Abdelrahman Mohamed, Peili Chen, Laurel H. Carney, Junfeng Lu, Jinsong Wu, and Edward F. Chang. Dissecting neural computations in the human auditory pathway using deep neural networks for speech. *Nature Neuroscience*, 26 (12):2213–2225, 2023. doi: 10.1038/s41593-023-01468-4.

[96] Greta Tuckute, Jenelle Feather, Dana Boebinger, and Josh H. McDermott. Many but not all deep neural network audio models capture brain responses and exhibit correspondence between model stages and brain regions. *PLOS Biology*, 21(12):e3002366, 2023. doi: 10.1371/journal.pbio.3002366.

[97] Ariel Goldstein, Haocheng Wang, Leonard Niekerken, Mariano Schain, Zaid Zada, Bobbi Aubrey, Tom Sheffer, Samuel A. Nastase, Harshvardhan Gazula, Aditi Singh, Aditi Rao, Gina Choe, Catherine Kim, Werner Doyle, Daniel Friedman, Sasha Devore, Patricia Dugan, Avinatan Hassidim, Michael Brenner, Yossi Matias, Orrin Devinsky, Adeen Flinker, and Uri Hasson. A unified acoustic-to-speech-to-language embedding space captures the neural basis of natural language processing in everyday conversations. *Nature Human Behaviour*, 9(5): 1041–1055, 2025. doi: 10.1038/s41562-025-02105-9.

[98] John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2733–2743, Hong Kong, China, 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1275.

[99] OpenAI. Whisper large-v3 model card. https://huggingface.co/openai/whisper-large-v3, 2023. Released November 2023 as `openai/whisper` `v20231106`. Accessed 11 September 2026.

[100] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. LibriSpeech: an ASR corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210. IEEE, 2015. doi: 10.1109/ICASSP.2015.7178964.

[101] Loren Lugosch. LibriSpeech alignments. Zenodo, version 1.0, doi:10.5281/zenodo.2619474, 2019. Word and phoneme alignments for LibriSpeech generated with the Montreal Forced Aligner and its pre-trained LibriSpeech acoustic model.

[102] Michael McAuliffe, Michaela Socolof, Sarah Mihuc, Michael Wagner, and Morgan Sonderegger. Montreal Forced Aligner: Trainable text-speech alignment using Kaldi. In *Proceedings of Interspeech 2017*, pages 498–502, 2017. doi: 10.21437/Interspeech.2017-1386.

[103] George A. Miller and Patricia E. Nicely. An analysis of perceptual confusions among some English consonants. *The Journal of the Acoustical Society of America*, 27(2):338–352, 1955. doi: 10.1121/1.1907526.

[104] John S. Garofolo, Lori F. Lamel, William M. Fisher, Jonathan G. Fiscus, David S. Pallett, Nancy L. Dahlgren, and Victor Zue. TIMIT acoustic-phonetic continuous speech corpus LDC93S1. Linguistic Data Consortium, Philadelphia. Web download, 1993.

[105] Kai-Fu Lee and Hsiao-Wuen Hon. Speaker-independent phone recognition using hidden Markov models. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(11):1641–1648, 1989. doi: 10.1109/29.46546.

[106] Steven Bird, Ewan Klein, and Edward Loper. *Natural Language Processing with Python*. O'Reilly Media, Sebastopol, CA, 2009.

[107] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *International Conference on Learning Representations, Workshop Track*, 2013. arXiv:1301.3781.

[108] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the*

2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423.

[109] Albert J Wakhloo, Tamara J Sussman, and SueYeon Chung. Linear classification of neural manifolds with correlated variability. *Physical Review Letters*, 131(2):027301, 2023.

[110] Jonathan D Wallis, Kathleen C Anderson, and Earl K Miller. Single neurons in prefrontal cortex encode abstract rules. *Nature*, 411(6840):953–956, 2001.

[111] Loren Lugosch, Mirco Ravanelli, Patrick Ignoto, Vikrant Singh Tomar, and Yoshua Bengio. Speech model pre-training for end-to-end spoken language understanding. In *Proceedings of Interspeech 2019*, pages 814–818, 2019.