Dimensions of Complexity Under Neural Compression

Niko Lorantos
University of Massachusetts Amherst

Abstract

Neural compression is fundamental to cognition, enabling intelligence to emerge in biological systems. Mapping high-dimensional data to latent representations requires compression, transforming structural complexity along distinct dimensions. We investigate how two distinct measures of complexity, effective rank and Kolmogorov complexity, transform under neural autoencoder compression. We analyze encoding mechanisms, examine architectural properties that enable complexity preservation, and consider implications of our work in machine learning and cognitive science. Through theoretical analysis, we characterize how different dimensions of structural complexity can be preserved through compression and conditions in which compression necessarily degrades complexity. This reveals how learned representations retain complex structure, with implications for understanding both artificial and biological neural systems.

1 Introduction

Compression is fundamental to cognition, enabling intelligence to emerge in biological systems constrained by neural bandwidth and homeostasis. Yet mapping high-dimensional data to low-dimensional representations necessarily transforms structure. Autoencoders are a class of neural architectures that encode high-dimensional inputs into low-dimensional latent representations, then reconstruct the original data with minimal information loss. This process of latent encoding informs our understanding of machine learning theory and brain processes such as efficient neural coding and cognitive abstraction.

Compression necessarily transforms structure though complexity itself is multidimensional rather than scalar. This work investigates how structural complexity is preserved through neural compression by analyzing two orthogonal measures: effective rank, which quantifies geometric dimensionality, and Kolmogorov complexity, which measures algorithmic regularity. Compressed representations may preserve these notions of structure differently, as a simple representation along one dimension may be intricate along another. This raises a fundamental question: how can the structural complexity of information be preserved through compression?

We guide our theoretical exploration with the following questions:

RQ1: When do effective rank and Kolmogorov complexity yield contradictory complexity assessments through neural compression?

RQ2: What mechanisms enable independent transformation of geometric and algorithmic complexity through nonlinear compression?

RQ3: What constraints determine complexity preservation in neural compression, and what does this reveal about learned representations?

2 Methods

2.1 Autoencoder Architecture

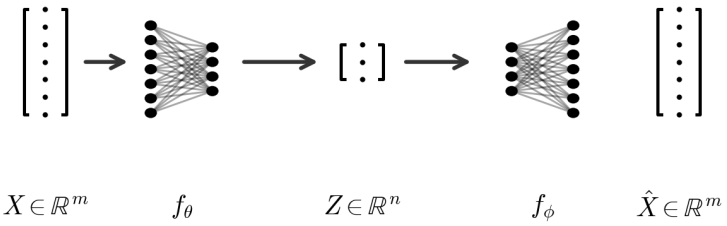

Figure 1: Autencoder architecture visualized.

An autoencoder maps high-dimensional inputs $X \in R^m$ to low-dimensional latent codes $Z \in R^n$ (where $n \ll m$) and back to $\hat{X} \in R^m$. To accomplish this, X is passed through an encoder, which returns a latent Z to be passed through the decoder to reconstruct $\hat{X} \in R^m$.

The autoencoder is then trained to minimize reconstruction loss:

$$\mathcal{L}(\theta, \phi) = \|X - \hat{X}\|^2 \tag{1}$$

where θ and ϕ are the learnable parameters of the encoder and decoder neural networks, respectively.

Autoencoders are high performers across tasks involving data compression and reconstruction[1]. Encoders act as feature extractors that capture irreducible structure, serving as a foundation of modern deep learning. Furthermore, autoencoders are effective anomaly detectors by identifying outliers through high reconstruction error. Autoencoders are a valuable tool across fields from representation learning to generative modeling.

[1]Bank, Dor, Noam Koenigstein, and Raja Giryes. "Autoencoders." Machine learning for data science handbook: data mining and knowledge discovery handbook (2023): 353-374.

2.2 Complexity

Complexity science studies systems with emergent collective behavior. This spans from weather patterns and bird flocking behavior to neural networks and consciousness. Central to this field is recognizing complexity as a multi-dimensional concept unable to be represented by a scalar value. Consider how a compressed representation of information might be simple along one dimension while remaining intricate along another. This multidimensionality is not a flaw, but rather a feature of complex systems' fundamental nature. The multidimensionality of complexity motivates this work to understand structure through compression as it may inform our understanding of information's intrinsic form.

We focus our analysis on structural complexity in particular, as the organizational properties of information are fundamental to preserve through compression for later reconstruction. To understand how compression transforms the structure of information, we focus our analysis on the encoding $f_\theta : R^m \to R^n$ from input to latent representation. We measure both input data and latent representations using two distinct measures of complexity outlined below.

2.2.1 Effective Rank

Effective rank, R, measures how many dimensions of a linear structure are meaningfully utilized to store information. For a matrix X with singular values $\sigma_1, \sigma_2, \ldots, \sigma_d$:

$$R(X) = \exp\left(-\sum_{i=1}^{d} p_i \log p_i\right) \tag{2}$$

where $p_i = \sigma_i / \sum_j \sigma_j$ normalizes each singular value. This quantifies how uniformly information is distributed across dimensions, or entropy of singular values. When singular values are concentrated in few dimensions, R is low. Effective rank ranges from 1 (when data lies in one dimension) to d (when data uniformly spans all dimensions), capturing the coordinative freedom and scale of geometric structure across dimensions.

2.2.2 Kolmogorov Complexity

Kolmogorov complexity K represents the length of the shortest program that generates a given string. Let us consider three example strings:

String	**Structure**	K
`abababab`	highly structured	low
`abcdezyxwv`	some structure	medium
`phsetivjwn`	unstructured	high

Table 1: K increases with randomness.

The first example has an obvious structure and can be reduced to $ab \times 5$, giving it low K. The second string represents one with "moderate" descriptive complexity. However, the third string is incompressible as it can only be minimally described verbatim, and thus has high K. These example help us to understand Kolmogorov complexity as intrinsic descriptive length[2].

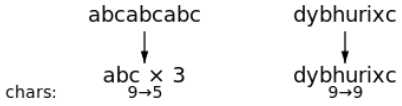

Figure 2: Minimal string descriptions with low K (left) and high K (right).

However, Kolmogorov complexity is uncomputable. To compute K would require running all possible programs and determining which shortest program generates a given string. This is impossible in practice as some programs will run forever due to the halting problem[3]. As K is uncomputable, compression-based approximation methods are often used in practice, such as measuring Lempel-Ziv complexity, bit-level entropy, or gzip compression size. These are sufficient for many practical cases, though we consider K abstractly for our theoretical analysis as we are interested in properties of K and R rather than specific computable approximations.

2.3 Approach

We conduct a theoretical analysis of how autoencoder compression transforms effective rank and Kolmogorov complexity. The analysis highlights a fundamental architectural property, that nonlinear encoding enables complexity to be stored explicitly in the latent representation Z or implicitly in learned encoder parameters θ.

We begin with linear compression as a baseline, then analyze how nonlinearity enable independent preservation of R and K. Finally, we identify the architectural and optimization constraints that bound successful complexity preservation.

3 Analysis

3.1 Linear Compression

Consider a linear encoding — no neural networks with learned parameters are used. Instead, information is directly compressed from X to Z. In such cases,

[2]Fortnow, Lance. "Kolmogorov complexity." Aspects of Complexity (Short Courses in Complexity from the New Zealand Mathematical Research Institute Summer 2000 Meeting, Kaikoura). Vol. 4. 2001.

[3]Turing, Alan Mathison. "On computable numbers, with an application to the Entscheidungsproblem." J. of Math 58.345-363 (1936): 5.

effective rank and Kolmogorov complexity are consistent; linear compression forces all complexity into latent coordinates. Thus, geometric dimensionality and descriptive structure are necessarily reduced to the bottleneck dimension n. An example of linear encoding is Principal Component Analysis (PCA), a popular statistical learning technique which finds the optimal linear encoder for high dimensional data. PCA demonstrates that for linear encoding, both R and K are determined by the same singular values spectrum and directly capture the distribution and its entropy, respectively. Hence, any linear compression necessarily reduces both measures of complexity to the latent representation Z.

3.2 Nonlinear Compression Mechanisms

Nonlinear autoencoders use neural networks for encoding and decoding, introducing a new mechanism for preserving complexity through compression. Nonlinear autoencoders decouple R and K as a result of the nonlinear activation functions of neurons within f_θ. Without nonlinear activation functions, a deep neural network would effectively act as a linear model as the composition of linear functions is necessarily another linear function. However, nonlinear activation functions like ReLU allow neural networks to extract intricate features from data. This nonlinearity is responsible for much of deep learning's success as it enables networks to learn abstract high-dimensional features of data.

As with linear encoding, nonlinear encoders compress information from X to Z. Thus, Z can explicitly preserve information. However, nonlinearity enables regularities to be captured by the learned encoder network f_θ. This implicit information storage takes a very different form from that of the latent space, and may preserve algorithmic notions of complexity like K different to those captured in Z. The interplay of these two systems enables K and R to be preserved through compression with greater degrees of freedom. Patterns with low K can be learned into parameters θ, while Z may store intrinsic dimensionality R. Latent compression enforces $n \ll m$, bounding $R(Z) \leq \min(n, R(X))$ and exploiting algorithmic structure through learned parameters θ.

3.3 Preserving Structural Complexity

Nonlinear autoencoders can preserve different dimensions of complexity through distinct mechanisms. The key insight is that complexity can either be stored explicitly in the latent representation Z, or implicitly in the learned parameters θ. This enables different dimensions of complexity to be preserved among different parts of the autoencoder architecture. We consider below how each structural complexity measure might be partitioned.

3.3.1 Geometric Structure

To understand how R can be preserved through compression, we must first understand the geometry of high-dimensional data. A *manifold* is a space that locally resembles Euclidean space even if its global structure is more complex.

A manifold can be thought of as locally flat, such that it behaves like standard Euclidean space in some dimension d. This dimension d is the manifold's intrinsic dimensionality. Hence, An example is Earth's surface, a 2D manifold despite being a sphere in 3D space.

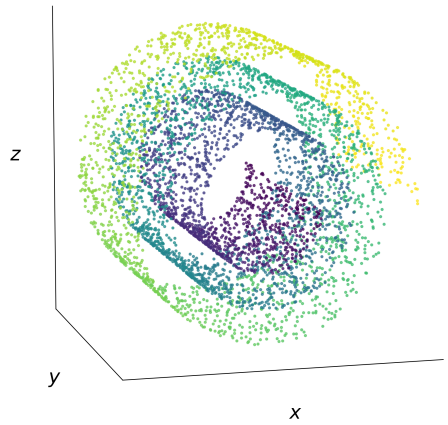

Figure 3: Swiss roll manifold, where local 2D Euclidean space is curved in 3D.

Consider the Swiss roll manifold in Figure 3, a 2D surface represented in 3D. Although only two coordinates are needed to specify position on the surface, the curvature of the space causes the representation to use all three dimensions. Linear compression methods like PCA cannot compress such a structure without extreme information loss. Projecting the Swiss roll linearly would collapse the spiral's layers, destroying the very structure necessary to preserve for learning.

Nonlinear encoders, however, can unfold curved manifolds where R reflects intrinsic structure rather than input space dimensionality. f_θ can learn to unroll the Swiss roll due to nonlinearity, achieving $R(Z) \approx 2$ while preserving geometric form. In this example, the mechanism succeeds when the latent dimension $n \geq 2$, large enough to accommodate the manifold's intrinsic geometry, allowing the compressed representation Z to capture intrinsic geometric dimensionality while parameters in f_θ learn regularities that are descriptive of the structural form.

3.3.2 Algorithmic Regularity

Recall that Kolmogorov complexity measures the length of the shortest program that generates a given string. When data possesses exploitable regularity (patterns that can be captured algorithmically), the encoder can learn these rules into θ rather than storing their output in Z. The latent representation

then becomes parameterized coordinates over learned structure rather than a raw encoding of the data itself.

An autoencoder can learn to decompose structure such that shared information is absorbed into the encoder's parameters θ, while the latent Z stores only the instance-specific information. This achieves $K(Z) < K(X)$ because individual latent vectors no longer encode the full specification. The compressed structure that reconstructs data now resides implicitly in the learned weights and captures regularities across the dataset.

This mechanism fundamentally depends on $K(X) < |X|$, meaning compressible structure exists for f_θ (and inversely f_ϕ) to learn. For unstructured or random data where $K(X) \approx |X|$, no such regularity exists and thus each sample is incompressible. In this case, the encoder must store information explicitly in Z, maintaining $K(Z) \approx K(X)$ though subject to information loss imposed by $n \ll m$.

3.4 Limitations

3.4.1 Latent Dimension

The relationship between the input dimension m, latent dimension n, and intrinsic dimensionality d determines how complexity is transformed through compression:

(*i*) $n < d$: Information loss is necessary as the latent representation cannot preserve all dimensions of intrinsic structure.

(*ii*) $n \approx d$: Optimal for nonlinear dimensionality reduction, as $R(Z)$ can preserve geometric structure while patterns may extract to θ.

(*iii*) $n \gg d$: Susceptible to overfitting as overparameterization eliminates pressure to learn generalizable regularities.

3.4.2 Learning Dynamics

The autoencoder loss function $\mathcal{L}(\theta, \phi)$ optimizes to minimize reconstruction error, blind to the preservation of structural complexity through compression. This indirect relationship between optimization and preserving complexity offers insight into how autoencoders preserve structure in practice.

The latent dimension imposes the constraint $R(Z) \leq n$, but does not inform how effectively geometric structure is preserved within this bound. When $n \geq d$, the encoder has sufficient capacity to unfold curved manifolds while preserving intrinsic geometry, with $R(Z)$ bounded by $d \leq R(Z) \leq n$. Achieving $R(Z) \approx (d)$ requires either explicit regularization or implicit pressure from optimization dynamics. In representations aimed to minimize reconstruction error, the loss function drives geometric preservation indirectly.

Descriptive complexity operates differently as it is implicitly stored through learned weights in f_θ when pattern extraction improves reconstruction. However, still only reconstruction loss is considered. Thus, in optimization settings,

patterns are learned because they minimize $\mathcal{L}(\theta, \phi)$, not inherently because they simplify the representation.

This reveals the fundamental asymmetry in how autoencoder compression transforms complexity. The loss function determines what structure is preserved based on reconstruction utility, not intrinsic complexity. Thus, R can be preserved when the latent dimension accommodates intrinsic dimensionality and f_θ learns to unfold manifold structure, while K reduces only when pattern extraction to θ improves reconstruction. However, since complexity is preserved only when aiding reconstruction, theoretical capacity limits are not necessarily realized through naive optimization of $\mathcal{L}(\theta, \phi)$ alone.

3.5 Addressing Research Questions

RQ1: When do effective rank and Kolmogorov complexity yield contradictory complexity assessments through neural compression?

Effective rank and Kolmogorov complexity yield contradictory assessments when nonlinear compression decouples geometric and algorithmic structure. This decoupling can be attributed to exploitable regularity ($K(X) < |X|$) on a manifold with intrinsic dimensionality d and latent dimension n satisfying $n \geq d$. Under these conditions, algorithmic patterns extract to learned parameters θ while intrinsic geometric is explicitly stored in Z, enabling algorithmic compression $K(Z) < K(X)$ while $R(Z) \approx d$. The orthogonal preservation of structural complexity emerges because descriptive complexity decreases through pattern extraction while geometric dimensionality is preserved through explicit storage enabled by nonlinearity within learned parameters θ.

Conversely, linear compression necessarily forces consistency. Any linear transformation maps singular value structure directly to both R and K, necessarily correlating preservation of the two complexity measures. Thus, independent preservation of both structural complexity notions requires nonlinearity as a necessary condition, with data structure and architectural capacity as sufficient conditions.

RQ2: What mechanisms enable independent transformation of geometric and algorithmic complexity through nonlinear compression?

The fundamental mechanism enabling learned preservation across the autoencoder architecture is the decomposition of complexity across the latent representation Z and learned parameters θ. This partitioning of the autoencoder architecture, a result of nonlinearity, enables independent preservation of R and K.

Manifold unfolding allows R to reflect intrinsic geometric complexity rather than input dimensionality. When $n \geq d$, nonlinear activation functions enable f_θ to learn coordinate transformations that unwrap curved manifolds, preserving geometric form despite dimensional reduction. This operates on the explicit representation Z, storing the unfolded manifold representation.

For algorithmic complexity, pattern extraction absorbs shared regularities into encoder weights. When $K(X) < |X|$, recurring descriptive structure generalizes into θ while Z stores only instance-specific parameters over this learned structure. This achieves $K(Z) < K(X)$ through implicit encoding as the algorithmic structure resides in the network weights rather than the latent space.

RQ3: What constraints determine complexity preservation in neural compression, and what does this reveal about learned representations?

Complexity preservation through neural compression is governed by architectural capacity and optimization objectives. These constraints differ as architectural capacity determines theoretical limits while optimization objectives determine practical realization of these bounds.

Capacity constraints establish necessary conditions for preservation. Geometric complexity requires $n \geq d$, ensuring sufficient latent dimensionality to accommodate intrinsic manifold structure. Algorithmic complexity requires $K(X) < |X|$, ensuring exploitable regularity exists in the data. When these conditions fail, total preservation of information becomes impossible regardless of optimization techniques; a latent space $n < d$ cannot fully represent intrinsic geometry d, and incompressible data with $K(X) \approx |X|$ offers no generalizability.

However, satisfying these capacity constraints does not guarantee preservation either. The reconstruction objective $\mathcal{L}(\theta, \phi) = \|X - \hat{X}\|^2$ drives complexity preservation only when it improves reconstruction. Geometric structure unfolds through f_θ because such representations minimize reconstruction error, not because they preserve intrinsic dimensionality. Similarly, patterns extract to θ because shared regularities reduce loss across the dataset, not because they simplify individual representations. This loss-centric optimization process is blind to complexity preservation, treating it as a mere byproduct of reconstruction utility.

This reveals learned representations as fundamentally task-centric rather than structure-centric. Autoencoders discover minimally sufficient encodings that preserve only the structural dimensions necessary for their objective function. The decoupling of R and K demonstrates that different tasks may require preserving different structures, with the optimization objective guiding the learning process. Hence, representation learning is not about capturing intrinsic data structure but about discovering task-relevant structure. Intelligence, from this view, emerges from representations that are minimally complex yet structured in ways that enable reuse.

4 Discussion

We now situate our findings of complexity transformation through neural compression within existing literature in machine learning and cognitive science. We specify mechanisms behind the manifold hypothesis and illuminate what makes for useful representations, then explore whether similar compression principles shape biological neural coding and abstraction.

4.1 Machine Learning

4.1.1 The Manifold Hypothesis

The manifold hypothesis theorizes that natural high-dimensional data lies near low-dimensional manifolds embedded within high-dimensional space.[4] This explains why dimensionality reduction succeeds in practice and motivates modern deep learning. However, while the hypothesis regards the existence of manifolds, it does not specify mechanisms that may enable their recovery through compression.

Our analysis identifies that manifold preservation requires not only existence of intrinsic dimension d, but that the encoder utilizes its nonlinear capacity. The Swiss roll example illustrates this, as linear encoding achieves low R by destroying the manifold structure while nonlinearity enables learned transformations of curved structure to flat Euclidean space.

This suggests a potential limitation in practical optimization settings. As relying on loss during training prioritizes evaluation reconstruction accuracy over true geometric learning, explicitly learning to preserve structure (e.g., $R(Z)$ relative to intrinsic dimensionality) may prove to be advantageous. Reconstruction quality would come as a byproduct of successful geometric preservation $R(Z) \approx d$ through compression.

4.1.2 Representation Learning

Representation learning aims to discover encodings that extract meaningful features from data and enable effective task performance. Our analysis reveals the task-centric nature of learned representations, preserving only minimally sufficient complexity to later reconstruct. The reconstruction loss merely preserves dimensions of complexity that improve reconstruction rather than attempting to learn the intrinsic structure.

Furthermore, different tasks exploit various notions of structural complexity. Geometric tasks such as clustering benefit from high $R(Z)$, preserving distributed structure in latent coordinates. On the other hand, generative tasks benefit from low $K(Z)$, extracting patterns to neural network parameters. We find this to inform our understanding of how the same autoencoder trained with different loss functions may discover different minimally sufficient representations.

This task-dependent nature of useful representations may explain unstable disentanglement metrics in unsupervised learning, with research finding disentanglement scores to vary dramatically across random seeds and hyperparameters[5]. Our analysis suggests why: optimizing for reconstruction loss gives the

[4] Fefferman, Charles, Sanjoy Mitter, and Hariharan Narayanan. "Testing the manifold hypothesis." Journal of the American Mathematical Society 29.4 (2016): 983-1049.

[5] Locatello, Francesco, et al. "Challenging common assumptions in the unsupervised learning of disentangled representations." international conference on machine learning. PMLR, 2019.

encoder freedom in how it shares learned structure between Z and θ. Each training run discovers a different solution, with all achieving similar reconstruction while distributing structure differently across the neural architecture. Disentanglement metrics thus measure task-dependent structure rather than intrinsic compositionality.

This shifts how we evaluate representations. Rather than considering how much information is preserved, good representations consider *which* complexity dimensions to preserve and for what purpose? Useful representations preserve structure that composes well: high R enables geometric task performance, while patterns learned in θ enable generalization capabilities. Representations morph according to task structure, promoting flexible reuse of learned components.

Representation quality is fundamentally relational rather than intrinsic. No representation is universally "good" as quality depends on alignment between preserved complexity and task requirements. Deep learning succeeds by discovering representations that are minimally complex yet structured for compositional reuse across tasks. Intelligence emerges not from maximal information preservation, but from selective preservation of task-relevant structure.

4.2 Cognitive Science and Intelligence

4.2.1 Efficient Coding and Neural Compression

The efficient coding hypothesis[6] posits that neural systems maximize information transmission while minimizing metabolic cost. This principle has shaped neuroscience for decades. However, while efficient coding predicts compression, it does not specify which dimensions are compressed and preserved. Through our theoretical analysis, we find structural complexity of different forms to be learnable by different components of neural compression systems, offering insight into structural preservation through efficient neural coding mechanisms in the brain.

Biological systems face metabolic constraints, and optimally preserve complexity with respect to its relevance as seen with autoencoder optimization techniques. If biological neural systems preserve structure in a similar manner to their artificial counterparts, our analysis predicts they should preserve geometric structure supporting spatial cognition while extracting algorithmic regularities to synaptic firing patterns. Furthermore, metabolic regulation creates pressure to learn efficient relations, implicitly driving structural preservation akin to the autoencoder loss function.

Our findings position metabolic efficiency as indirect pressure to compress and preserve only the necessary structure that contributes to reconstruction. Natural sensory input to biological brains often contain high-dimensional structure with exploitable regularities. Systems under metabolic constraints cannot preserve all structural dimensions, thus having to make the same tradeoffs as autoencoders with limited capacity. As with autoencoders, it may be the case

[6]Barlow, Horace B. "Possible principles underlying the transformation of sensory messages." Sensory communication 1.01 (1961): 217-233.

that the optimal strategy is dependent on the reconstruction task. We expect selective compression across neural systems to follow from our analysis. Different brain regions likely sacrifice different dimensions of complexity based on practical necessity. Efficient coding can be understood as the selective encoding process where systems preserve specific structure rather than maximizing information retention.

Biological systems minimize their objective combining reconstruction accuracy with metabolic cost proportional to neural firing activity. This creates architectural pressure toward sparse activity patterns and stable connection weights, analogous to the latent dimension of the autoencoder architecture. Importantly, neural dynamics operate on asynchronous timescales. As a result, rapid neural activity encodes instance-specific information while Hebbian learning of synaptic plasticity reinforces recurring patterns over time. In practice, this timescale separation may realize the decomposition of learned structure among neural weights and latent information. If this framework holds, we predict:

1. Neural representations should exhibit task-dependent complexity preservation, transforming structure differently across brain regions (i.e., specialized regions for geometric vs. predictive coding)

2. Hebbian learning should encode regularities into connection weights

3. Metabolic regulation should correlate with learned synaptic structure, bounding practical realization of learned structural preservation

Though empirical validation of these possible neuroscientific implications extends beyond the scope of this work, our theoretical framework offers an explanation of how efficient coding might preserve complexity of information within the brain.

4.2.2 Abstraction and Hierarchical Learning

Abstraction is often described as a process of simplifying cognitive thought by discarding specifics and reducing information to its generalized form[7]. Our analysis gives mathematical form to this intuition, demonstrating how abstraction and algorithmic compression alike share structure in learned representations, preserving essential features while discarding unnecessary details.

Consider how abstraction operates as a computational mechanism. Rather than encoding complete high-dimensional data, abstract representations become regularity over learned structure. For example, the string "abababababab" with low K reduces effectively through pattern extraction to θ. In this example, algorithmic structure resides in the network weights rather than the latent representation itself; abstract representations similarly compress information with respect to descriptive complexity.

[7] "Abstraction." EBSCO Research Starters, www.ebsco.com/research-starters/religion-and-philosophy/abstraction. Accessed 15 Nov. 2025.

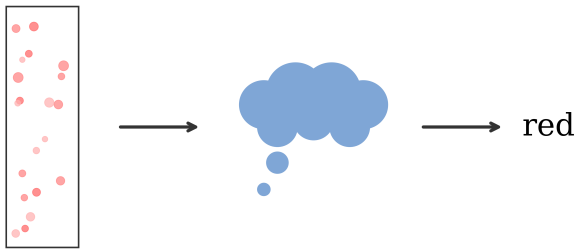

Figure 4: An illustration of cognitive abstraction.

Systems capable of learning abstractions utilize this mechanism when exploitable regularity exists ($K(X) < |X|$). As the encoder discovers recurring patterns, they can be compressed into neural weights to reduce representational complexity. Furthermore, the latent Z may store only the minimal parameters needed to specify the encoding instance. This relation parallels the interplay of working memory and learned pattern recognition in human cognition.

Hierarchical abstraction emerges from recursive abstraction. Each phase (or layer) extracts structure from the prior phase, storing them in nonlinear space. This composition enables progressively abstract representations. Early layers might extract local features while deeper layers extract compositional patterns, aligning with how convolutional neural networks extract relevant features through deep learning[8].

Few-shot learning, that is, learning from a limited number of samples, becomes tractable when considering representations with low K structure only requiring further knowledge instance-specific parameters over pre-existing structure. Furthermore, skill and knowledge transfer across tasks is likely to succeed when domains share structural properties. However, useful abstraction again requires $K(X) < |X|$, as random unstructured data offers no regularity to learn.

Hence, intelligence emerges from discovering which dimensions of complexity to preserve for a given task. This goes against the notion that intelligence correlates with maximal information, instead suggesting its correlation with minimally sufficient information.

5 Conclusion

Using autoencoders as the subject of theoretical analysis, we explore how structural complexity transforms under neural compression, analyzing two orthogonal

[8]Li, Zewen, et al. "A survey of convolutional neural networks: analysis, applications, and prospects." IEEE transactions on neural networks and learning systems 33.12 (2021): 6999-7019.

measures: effective rank and Kolmogorov complexity. These notions of complexity measure geometric dimensionality and algorithmic regularity of data, respectively. Our analysis reveals the fundamental preservation mechanisms of linear and nonlinear compression. Linear encoding necessarily reduces both R and K proportionally, forcing all complexity into the latent representation Z. Nonlinear encoding, however, enables independent transformation of these complexity dimensions as geometric structure can be preserved explicitly in Z while exploitable regularity is implicitly learned into the parameters θ. This compression and preservation of structure highly depends on algorithmic compressibility, architectural capacity, and optimization dynamics.

Our findings frame learning as fundamentally goal-oriented, preserving structure of information only when it improves performance. Essential to intelligence is the process of discovering which complexity dimensions to preserve for which tasks. This principle extends beyond artificial neural networks, offering explanations for cognitive abstraction and efficient neural coding in biological systems where metabolic constraints drive selective preservation of relevant structure. Learned representations are thus minimally complex yet structured in ways that enable reuse.

Effective compression learns which dimensions of information are necessary for reconstruction. The implications of this selective preservation of complexity positions general intelligence as the ability to discover task-relevant structure rather than preserve maximal information.