Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Position Paper:

On the principles of Parsimony and Self-consistency for the emergence of intelligence

Yi MA[‡1], Doris TSAO[2], Heung-Yeung SHUM[3]

[1]*Electrical Engineering and Computer Science Department, University of California, Berkeley, CA 94720, USA*

[2]*Department of Molecular & Cell Biology and Howard Hughes Medical Institute, University of California, Berkeley, CA 94720, USA*

[3]*International Digital Economy Academy, Shenzhen 518045, China*

E-mail: yima@eecs.berkeley.edu; dortsao@berkeley.edu; hshum@idea.edu.cn

Received July 10, 2022; Revision accepted July 24, 2022; Crosschecked July 29, 2022; Published online Aug. 12, 2022

Abstract: Ten years into the revival of deep networks and artificial intelligence, we propose a theoretical framework that sheds light on understanding deep networks within a bigger picture of intelligence in general. We introduce two fundamental principles, *Parsimony* and *Self-consistency*, which address two fundamental questions regarding intelligence: what to learn and how to learn, respectively. We believe the two principles serve as the cornerstone for the emergence of intelligence, artificial or natural. While they have rich classical roots, we argue that they can be stated anew in entirely measurable and computable ways. More specifically, the two principles lead to an effective and efficient computational framework, compressive closed-loop transcription, which unifies and explains the evolution of modern deep networks and most practices of artificial intelligence. While we use mainly visual data modeling as an example, we believe the two principles will unify understanding of broad families of autonomous intelligent systems and provide a framework for understanding the brain.

Key words: Intelligence; Parsimony; Self-consistency; Rate reduction; Deep networks; Closed-loop transcription
https://doi.org/10.1631/FITEE.2200297 **CLC number:** TP18

1 Context and motivation

For an autonomous intelligent agent to survive and function in a complex environment, it must efficiently and effectively learn models that reflect both its past experiences and the current environment being perceived. Such models are critical for gathering information, making decisions, and taking action. Generally referred to as world models, these models should be continuously improved based on how projections agree with new observations and outcomes. They should incorporate both knowledge from past experiences (e.g., recognizing familiar objects) and mechanisms for interpreting immediate sensory inputs (e.g., detecting and tracking moving objects). Studies in neuroscience suggest that the brain's world model is highly *structured* anatomically (e.g., modular brain areas and columnar organization) and functionally (e.g., sparse coding (Olshausen and Field, 1996) and subspace coding (Chang and Tsao, 2017; Bao et al., 2020)). Such a structured model is believed to be the key to the brain's efficiency and effectiveness in perceiving, predicting, and making intelligent decisions (Barlow, 1961; Josselyn and Tonegawa, 2020).

In contrast, in the past decade, progress in artificial intelligence has relied mainly on training "tried-and-tested" models with largely homogeneous structures, like deep neural networks (DNNs) (LeCun et al., 2015), using a brute-force engineering approach. While functional modularity may emerge

‡ Corresponding author
ORCID: Yi MA, https://orcid.org/0000-0001-5485-419X
© Zhejiang University Press 2022

from training, the learned feature representation remains largely hidden or latent inside and is difficult to interpret (Zeiler and Fergus, 2014). Currently, such expensive brute-force end-to-end training of black-box models has resulted in ever-growing model size and high data/computation cost[1], and is accompanied by many caveats in practice: the lack of richness in final learned representations due to neural collapse (Papyan et al., 2020)[2], lack of stability in training due to mode collapse (Srivastava A et al., 2017), lack of adaptiveness and susceptibility to catastrophic forgetting (McCloskey and Cohen, 1989), and lack of robustness to deformations (Azulay and Weiss, 2019; Engstrom et al., 2019) or adversarial attacks (Szegedy et al., 2014).

A principled and unifying approach? We hypothesize that a fundamental reason why these problems arise in the current practice of deep networks and artificial intelligence is a lack of systematic and integrated understanding about the functional and organizational principles of intelligent systems.

For instance, training discriminative models for classification and generative models for sampling or replaying has been largely separated in practice. Such models are typically open-loop systems that must be trained end to end via supervision or self-supervision. A principle long-learned in control theory is that such open-loop systems cannot automatically correct errors in prediction, and are unadaptive to changes in the environment. This had led to the introduction of "closed-loop feedback" to controlled systems so that a system can learn to correct its errors (Wiener, 1948; Mayr, 1970). As we will argue in this paper, a similar lesson can be drawn here: once discriminative and generative models are combined to form a complete closed-loop system, learning can become autonomous (without exterior supervision), more efficient, stable, and adaptive.

To understand any functional component that may be necessary for an intelligent system, such as a discriminative or a generative segment, we need to understand intelligence from a more principled and unifying perspective. Therefore, in this paper, we introduce two fundamental principles, *Parsimony* and *Self-consistency*, which we believe govern the function and design of any intelligent system, artificial or natural. The two principles aim to answer the following two fundamental questions regarding learning, respectively:

1. *What to learn:* what is the objective of learning from data, and how can it be measured?

2. *How to learn:* how can we achieve such an objective via efficient and effective computation?

As we will see, answers to the first question fall naturally into the realm of information/coding theory (Shannon, 1948), which studies how to accurately *quantify and measure* the information in the data and then seek *the most compact* representations of the information. Once the objective of learning is clear and set, answers to the second question fall naturally into the realm of control/game theory (Wiener, 1948), which provides a universally effective computational framework, i.e., a *closed-loop* feedback system, for achieving any measurable objective *consistently* (Fig. 1).

The basic ideas behind each of the two principles proposed in this paper can find their roots in classic works. Artificial (deep) neural networks, since their earliest inception as "perceptrons" (Rosenblatt, 1958), were conceived to store and organize sensory information efficiently. Back propagation (Kelley, 1960; Rumelhart et al., 1986) was later proposed as a mechanism for learning such models. Moreover, even before the inception of neural networks, Norbert Wiener had contemplated computational mechanisms for learning at a system level. In his famed book *Cybernetics* (Wiener, 1961), he studied the possible roles of information compression for parsimony and feedback/games in a learning machine for consistency.

But we are here to reunite and restate the two principles within the new context of data science and machine learning, as they help better explain and unify most modern instances and practices of artificial intelligence, particularly deep learning[3].

[1]With model sizes frequently going beyond billions or trillions of parameters, even Google seems to recently have started worrying about the carbon footprint of such practices (Patterson et al., 2022).

[2]This refers to the final representation for each class collapsing to a one-hot vector that carries no information about the input except its class label. Richer features might be learned inside the networks, but their structures are unclear and remain largely hidden.

[3]As we will see, besides integrating discriminative and generative models, they lead to a closed-loop framework that works uniformly in supervised, incremental, or unsupervised settings, without suffering from many of the problems of open-loop deep networks.

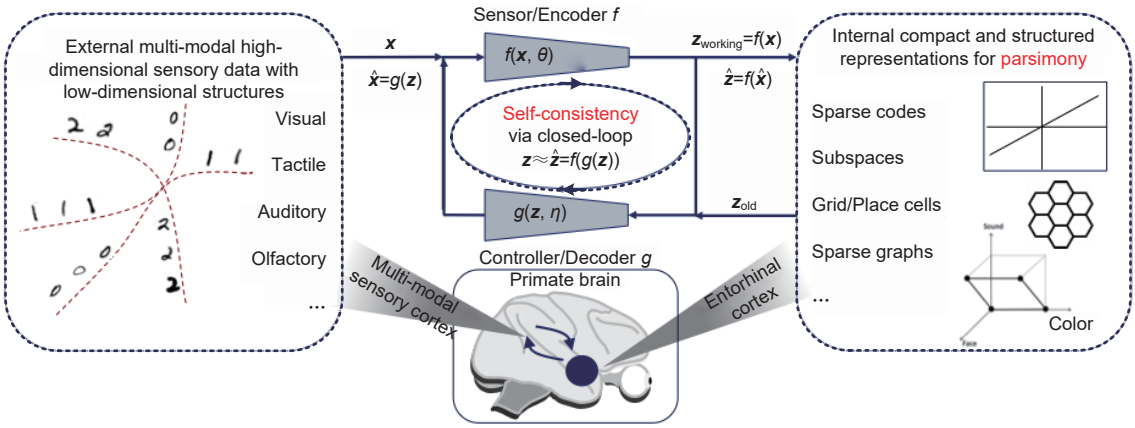

Fig. 1 Overall framework for a universal learning engine. We seek a compact and structured model for sensory data via a compressive closed-loop transcription. It includes a (nonlinear) mapping $f(\cdot,\theta): x \mapsto z$ that maps high-dimensional sensory data with complicated low-dimensional structures to a compact structured representation. The model needs to be self-consistent; i.e., it can regenerate the original data via a map $g(\cdot,\eta): z \mapsto \hat{x}$ such that f cannot distinguish despite its best effort. Often, for simplicity, we omit the dependency of the mappings f and g on their parameters θ and η, respectively.

Different from previous efforts, our restatement of these principles will be entirely *measurable* and *computationally tractable*, hence easily realizable by machines or in nature with limited resources. This paper aims to offer our overall position and perspective rather than to justify every claim technically. Nevertheless, we will provide references to related work where readers can find convincing theoretical and compelling empirical evidence. They are based on a coherent series of past and recent developments in the study of machine learning and data science by the authors and their students (Ma et al., 2007; Wright et al., 2007; Chan TH et al., 2015; Yu YD et al., 2020; Baek et al., 2022; Chan KHR et al., 2022; Dai et al., 2022; Pai et al., 2022; Tong et al., 2022; Wright and Ma, 2022).

The rest of this paper is organized as follows: In Section 2, we use visual data modeling as a concrete example to introduce the two principles and illustrate how they can be instantiated as computable objectives, architectures, and systems. In Section 3, we conjecture that they lead to a universal learning engine for broader perception and decision making tasks. In Section 4, we discuss several implications of the proposed principles and their connections to neuroscience, mathematics, and higher-level intelligence. Finally, Section 5 concludes the paper.

2 Two principles for intelligence

In this section, we introduce and explain the two fundamental principles that can help answer the questions of *what to learn* and *how to learn* by an intelligent agent or system.

2.1 What to learn: the principle of Parsimony

"*Entities should not be multiplied unnecessarily.*"
— William of Ockham

The principle of Parsimony: *The objective of learning for an intelligent system is to identify low-dimensional structures in observations of the external world and reorganize them in the most compact and structured way.*

There is a fundamental reason why intelligent systems need to embody this principle: intelligence would be impossible without it! If observations of the external world had no low-dimensional structures, nothing would be worth learning or memorizing. Nothing could be relied upon for good generalization or prediction, which relies on new observations following the same low-dimensional structures. Thus, this *principle* is not simply a *convenience* arising from the need for intelligent systems to be frugal with their resources, such as energy, space, time, and matter.

In some contexts, the above principle is also

called the *principle of Compression*. However, *Parsimony* of intelligence is not about achieving the best possible compression, but about *obtaining compact and structured representations via computationally efficient means*. There is no point for an intelligent system to try to compress data to the ultimate level of Kolmogorov complexity or Shannon information: they not only are intractable to compute (or even to approximate) but also result in completely unstructured representations. For instance, representing data with the minimum description length (Shannon information) requires minimizing the Helmholtz free energy via a Helmholtz machine (Hinton et al., 1995), which is typically computationally intractable. When examined more closely, many commonly used mathematical or statistical "measures" for model goodness are either *exponentially expensive to compute* for general high-dimensional models or even become *ill-defined* for data distributions with low-dimensional supports. These measures include widely used quantities, such as maximum likelihood, Kullback–Leibler (KL) divergence, mutual information, and Jensen–Shannon and Wasserstein distances[4]. It is commonplace in the practice of machine learning to resort to various heuristic approximations and empirical evaluations. As a result, performance guarantees and understanding are lacking.

Now we face a question: how can an intelligent system embody the *principle of Parsimony* to identify and represent structures in observations in a computationally tractable and even efficient way? Theoretically, an intelligent system could use any family of desirable structured models for the world, provided that they are simple yet expressive enough to model informative structures in real-world sensory data. The system should be able to accurately and efficiently evaluate how good a learned model is, and the measure used should be basic, universal, and tractable to compute and optimize. What is a good choice for a family of structured models with such a measure?

To see how we can model and compute parsimony, we use the motivating and intuitive example of modeling visual data[5]. To make our exposition easy, we will start with a supervised setting in this section. Nevertheless, as will be discussed in Section 2.2, with parsimony as the only "self-supervision" and with the second principle of Self-consistency, a learning system can become fully autonomous and function without needing any exterior supervision.

2.1.1 Modeling and computing parsimony

Let us use $\boldsymbol{x}$ to denote the input sensory data (e.g., an image), and $\boldsymbol{z}$ its internal representation. The sensory data sample $\boldsymbol{x} \in \mathbb{R}^D$ is typically rather high-dimensional (millions of pixels) but has extremely low-dimensional intrinsic structures[6]. Without loss of generality, we may assume that it is distributed on some low-dimensional submanifolds, as illustrated in Fig. 2. Then, the purpose of learning is to establish a (usually nonlinear) mapping f, say in some parametric family $\theta \in \Theta$, from $\boldsymbol{x}$ to a much lower-dimensional representation $\boldsymbol{z} \in \mathbb{R}^d$:

$$\boldsymbol{x} \in \mathbb{R}^D \xrightarrow{f(\boldsymbol{x},\theta)} \boldsymbol{z} \in \mathbb{R}^d, \tag{1}$$

such that the distribution of feature $\boldsymbol{z}$ is much more compact and structured. Being compact means economic to store. Being structured means efficient to access and use. Particularly, linear structures are ideal for interpolation or extrapolation.

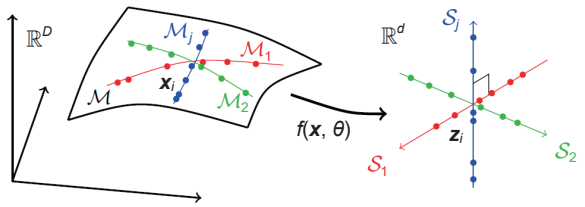

Fig. 2 Seeking a linear and discriminative representation: mapping high-dimensional sensory data, typically distributed on many nonlinear low-dimensional submanifolds, onto a set of independent linear subspaces of the same dimensions as the submanifolds.

To be more precise, we can formally instantiate the principle of *Parsimony* for visual data modeling as trying to find a (nonlinear) transform f that achieves the following goals:

1. Compression: map high-dimensional sensory data $\boldsymbol{x}$ to a low-dimensional representation $\boldsymbol{z}$;

[4]More explanations about caveats associated with these measures can be found in Ma et al. (2007) and Dai et al. (2022).

[5]It is arguably true that vision is the most complex to model among all senses.

[6]For example, all images of a rotating pen trace out only a one-dimensional curve in the space of millions of pixels.

2. Linearization: map each class of objects distributed on a nonlinear submanifold to a linear subspace;

3. Sparsification: map different classes into subspaces with independent or maximally incoherent bases[7].

In other words, we try to transform real-world data that may lie on a family of low-dimensional submanifolds in a high-dimensional space onto a family of independent low-dimensional linear subspaces. Such a model is called a *linear discriminative representation* (LDR) (Yu YD et al., 2020; Chan KHR et al., 2022), and the compression process is illustrated in Fig. 2. In some sense, one may even view the common practice of deep learning that maps each class to a "one-hot" vector as seeking a very special type of LDR models in which each target subspace is only one-dimensional and orthogonal to others.

The idea of compression as a guiding principle of the brain for representing (sensory data of) the world has strong roots in neuroscience, going back to Barlow's efficient coding hypothesis (Barlow, 1961). Scientific studies have shown that visual object representations in the brain exhibit compact structures, such as sparse codes (Olshausen and Field, 1996) and subspaces (Chang and Tsao, 2017; Bao et al., 2020). This supports the proposal that low-dimensional linear models are the preferred representations in the brain (at least for visual data).

2.1.2 Maximizing rate reduction

Remarkably, for the family of LDR models, there is a natural intrinsic measure of parsimony. Intuitively speaking, given an LDR, we can compute the total "volume" spanned by all features on all subspaces and the sum of "volumes" spanned by features of each class. Then the ratio between these two volumes gives a natural measure that suggests how good the LDR model is: the larger, the better. Fig. 3 shows an example with features distributed on two subspaces, S_1 and S_2. Models on the left and right have the same intrinsic complexity. The configuration on the left is preferred as features for different classes are made independent and orthogonal—their extrinsic representations would be the most sparse. Hence, in terms of this basic volumetric measure, the best representation should be such that "*the whole is maximally greater than the sum of its parts*."

As per information theory, the volume of a distribution can be measured by its *rate distortion* (Cover and Thomas, 2006). Roughly speaking, the rate distortion is the logarithm of how many ϵ-balls or spheres one can pack into the space spanned by a distribution[8]. The logarithm of the number of balls directly translates into how many binary bits one needs in order to encode a random sample drawn from the distribution subject to the precision ϵ. This is generally known as the *description length* (Rissanen, 1989; Ma et al., 2007).

Now let R be the rate distortion of the joint distribution of all features $\boldsymbol{Z} \doteq [\boldsymbol{z}^1, \boldsymbol{z}^2, \ldots, \boldsymbol{z}^n]$ of sampled data $\boldsymbol{X} \doteq [\boldsymbol{x}^1, \boldsymbol{x}^2, \ldots, \boldsymbol{x}^n]$ from all, say k, classes. R^{c} is the average of the rate distortions for the k classes: $R^{\mathrm{c}}(\boldsymbol{Z}) = \frac{1}{k}[R(\boldsymbol{Z}_1) + R(\boldsymbol{Z}_2) + \ldots + R(\boldsymbol{Z}_k)]$ where $\boldsymbol{Z} = \boldsymbol{Z}_1 \cup \boldsymbol{Z}_2 \cup \ldots \cup \boldsymbol{Z}_k$. Note that because of the logarithm, the ratio between volumes becomes the difference between rates. Then the difference between the whole and the sum of the parts, called the *rate reduction* (Chan KHR et al., 2022):

$$\Delta R(\boldsymbol{Z}) \doteq R(\boldsymbol{Z}) - R^{\mathrm{c}}(\boldsymbol{Z}), \tag{2}$$

gives the most basic, bean-counting-like, measure of how good the feature representation $\boldsymbol{Z}$ is[9].

Although for general distributions in high-dimensional spaces, the rate distortion, like many other measures mentioned before, is intractable and

[7]This is related to the notion of sparse dictionary learning (Zhai et al., 2020) or independent component analysis (ICA) (Hyvärinen, 1997; Hyvärinen and Oja, 1997). Once the bases of the subspaces are made independent or incoherent by the transform, the resulting representation becomes sparse and thus collectively compact and structured. For example, two sets of subspaces with the same dimensions have the same intrinsic complexity. However, their extrinsic representations can be very different (Fig. 3). This illustrates why simply compressing data based on their intrinsic complexity is insufficient for parsimony.

[8]Sphere packing gives almost a universal way to measure the volume of space of an arbitrary shape: to compare volumes of two containers, one only has to fill them both with beans and then count and compare. Optimal sphere-packing problems can be traced back to Johannes Kepler since 1611. Most recently, mathematician Maryna Viazovska received the 2022 Fields medal for solving the optimal sphere-packing problem in spaces of dimension 8 (Viazovska, 2017) and 24 (Cohen H et al., 2017).

[9]The rate reduction quantity also has a natural interpretation as "information gain" (Quinlan, 1986). It measures how much information is gained, in terms of bits saved, by specifying a sample on one of the parts, compared to drawing a random sample from the whole.

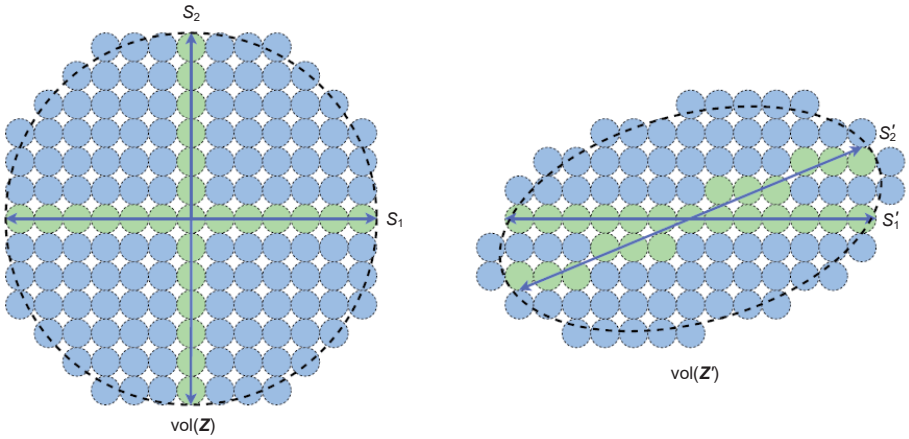

Fig. 3 Rate of all features $\boldsymbol{R} = \log\#$ (green spheres + blue spheres) and average rate of features on the two subspaces $\boldsymbol{R}^{\mathrm{c}} = \log\#$ (green spheres). Rate reduction is the difference between the two rates: $\boldsymbol{\Delta R} = \boldsymbol{R} - \boldsymbol{R}^{\mathrm{c}}$.

NP-hard to compute (MacDonald et al., 2019), the rate distortion for data $\boldsymbol{Z}$ drawn from a Gaussian distribution supported on a subspace has a closed-form formula (Ma et al., 2007):

$$R(\boldsymbol{Z}) \doteq \frac{1}{2}\log\big(\det\left(\boldsymbol{I} + \alpha \boldsymbol{Z}\boldsymbol{Z}^*\right)\big). \tag{3}$$

Hence, it can be efficiently computed and optimized!

The work of Chan KHR et al. (2022) has shown that if one uses the rate distortion functions of Gaussians and chooses a generic deep network (say a ResNet) to model the mapping $f(\boldsymbol{x}, \theta)$, then by maximizing the coding rate reduction, known as *the MCR^2 principle*:

$$\max_{\theta} \Delta R(\boldsymbol{Z}(\theta)) = R(\boldsymbol{Z}(\theta)) - R^{\mathrm{c}}(\boldsymbol{Z}(\theta)), \tag{4}$$

one can effectively map a multi-class visual dataset to multiple orthogonal subspaces. Notice that maximizing the first term of the rate reduction R expands the volume of all features. It simultaneously conducts "constrastive learning" for all features, which can be much more effective than contrasting sample pairs as normally conducted in conventional contrastive methods (Hadsell et al., 2006; van den Oord et al., 2019). Minimizing the second term R^{c} compresses and linearizes the features in each class. This can be interpreted as conducting "contractive learning" (Rifai et al., 2011) for each class. The rate reduction objective unifies and generalizes these heuristics.

Particularly, one can rigorously show that, by maximizing the rate reduction, features of different classes will be independent and features of each class will be distributed *almost uniformly* within each subspace (Chan KHR et al., 2022). In contrast, the widely practiced *cross entropy* objective for mapping each class to a one-hot label maps the final features of each class onto a one-dimensional singleton (Papyan et al., 2020).

2.1.3 White-box deep networks from unrolling optimization

Notice that in this context, the role of a deep network is simply to model the nonlinear mapping f between the external data $\boldsymbol{x}$ and the internal representation $\boldsymbol{z}$. How should an intelligent system know what family of models to use for the map f in the first place? Is there a way to directly derive and construct such a mapping instead of guessing and trying different possibilities?

Recall that our goal is to optimize the rate reduction $\Delta R(\boldsymbol{Z})$ as a function of the set of features $\boldsymbol{Z}$. To this end, we may directly start with the original data $\boldsymbol{Z}_0 = \boldsymbol{X}$ and incrementally optimize $\Delta R(\boldsymbol{Z})$, say with a *projected gradient ascent* (PGA) scheme[10]:

$$\boldsymbol{Z}_{\ell+1} \propto \boldsymbol{Z}_{\ell} + \eta \cdot \frac{\partial \Delta R}{\partial \boldsymbol{Z}}\bigg|_{\boldsymbol{Z}_{\ell}} \quad \text{subject to } \|\boldsymbol{Z}_{\ell+1}\| = 1. \tag{5}$$

That is, one can follow the gradient of the rate reduction to move the features so as to increase the rate reduction. Such a gradient-based iterative deformation process is illustrated in Fig. 4.

[10]For fair comparison of coding rates between two representations, we need to normalize the scale of the features, say $\|\boldsymbol{z}\| = 1$.

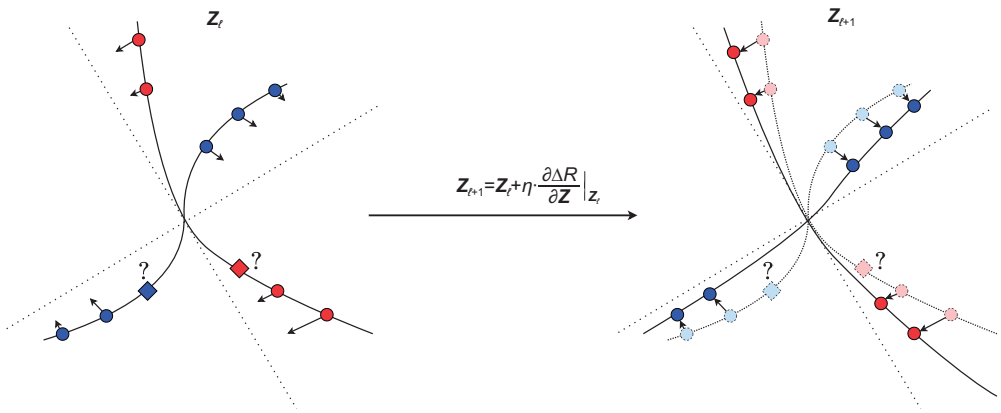

Fig. 4 A basic way to construct the nonlinear mapping f: following the local gradient flow $\frac{\partial \Delta R(Z)}{\partial Z}$ of the rate reduction ΔR, we incrementally linearize and compress features on nonlinear submanifolds and separate different submanifolds to respective orthogonal subspaces (the two dotted lines).

From the closed-form formula for the rate distortions (Eq. (3)), we can also compute the gradient of $\Delta R = R - R^{\mathrm{c}}$ in the closed form. For example, the gradient of the first term R is of the form

$$\left.\frac{\partial R(\boldsymbol{Z})}{\partial \boldsymbol{Z}}\right|_{\boldsymbol{Z}_\ell} = \frac{1}{2}\left.\frac{\partial \log\big(\det(\boldsymbol{I}+\alpha \boldsymbol{Z}\boldsymbol{Z}^*)\big)}{\partial \boldsymbol{Z}}\right|_{\boldsymbol{Z}_\ell} \tag{6}$$

$$= \alpha(\boldsymbol{I}+\alpha \boldsymbol{Z}_\ell \boldsymbol{Z}_\ell^*)^{-1}\boldsymbol{Z}_\ell \doteq \boldsymbol{E}_\ell \boldsymbol{Z}_\ell. \tag{7}$$

Similarly, we can compute the gradients for the k terms $\{R(\boldsymbol{Z}_i)\}_{i=1}^k$ in R^{c} and obtain k operators on $\boldsymbol{Z}_\ell$, named $\boldsymbol{C}_i$. Then, the above gradient ascent operation (5) takes the following structured form:

$$\boldsymbol{z}_{\ell+1} \propto \boldsymbol{z}_\ell + \eta\Big[\boldsymbol{E}_\ell \boldsymbol{z}_\ell + \boldsymbol{\sigma}\big([\boldsymbol{C}_\ell^1 \boldsymbol{z}_\ell, \boldsymbol{C}_\ell^2 \boldsymbol{z}_\ell, \ldots, \boldsymbol{C}_\ell^k \boldsymbol{z}_\ell]\big)\Big] \quad \text{subject to } \|\boldsymbol{z}_{\ell+1}\| = 1, \tag{8}$$

where $\boldsymbol{E}_\ell$ and $\boldsymbol{C}_\ell$'s are linear operators fully determined by covariances of the features from the previous layer $\boldsymbol{Z}_\ell$ (Eq. (7))[11]. Here, $\boldsymbol{\sigma}$ is a softmax operator that assigns $\boldsymbol{z}_\ell$ to its closest class based on its distance to each class, measured by $\boldsymbol{C}_\ell \boldsymbol{z}_\ell$. A diagram of all the operators per iteration is given in Fig. 5a.

Acute readers may have recognized that such a diagram draws a good resemblance to a layer of popular "tried-and-tested" deep networks such as ResNet (He et al., 2016) (Fig. 5b), including parallel columns as in ResNeXt (Xie et al., 2017) (Fig. 5c) and a mixture of experts (MoE) (Shazeer et al., 2017). This provides a natural and plausible interpretation of an important class of DNNs from the perspective of *unrolling an optimization scheme*. Even before the rise of modern deep networks, iterative optimization schemes for seeking sparsity, such as the iterative soft thresholding algorithm (ISTA) or fast ISTA (FISTA) (Wright and Ma, 2022), had been interpreted as learnable deep networks, e.g., the work of Gregor and LeCun (2010) on learned ISTA[12]. The class of networks derived from optimizing rate reduction has been named *ReduNet* (Chan KHR et al., 2022).

2.1.4 Forward unrolling versus backward propagation

We see above that compression leads to an entirely constructive way of deriving a DNN, including its architecture and parameters, as a fully interpretable *white-box*[13]: its layers conduct iterative and incremental optimization of a principled objective that promotes parsimony. As a result, for so-obtained deep networks, the ReduNets, starting from

[11] $\boldsymbol{E}$ is associated with the gradient of the first term R and stands for "expansion" of the whole set of features, whereas $\boldsymbol{C}$'s are associated with the gradients of multiple rate distortions in the second term R^{c} and stand for "compression" of features in each class. See Chan KHR et al. (2022) for the details.

[12] A strong connection between sparsity and deep convolutional neural networks (CNNs) was formally established by Papyan et al. (2018). Similarly, unfolding iterative optimization for sequential sparse recovery leads to recurrent neural networks (RNNs) (Wisdom et al., 2017).

[13] Here, we give only an interpretation of deep networks, instead of artificial intelligence in general, which, we believe, remains an open research topic, as we will discuss more in Section 4.

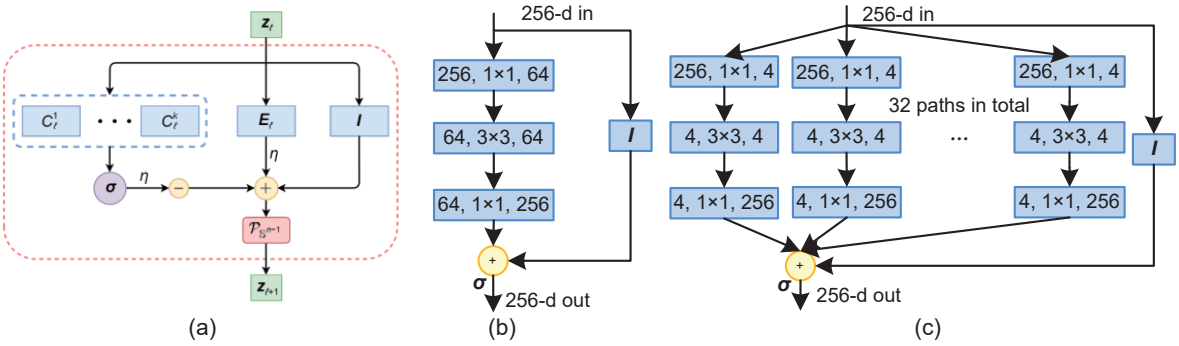

Fig. 5 Building blocks of the nonlinear mapping f: (a) one layer of ReduNet as one iteration of projected gradient ascent, which precisely consists of expansive or compressive linear operators, a nonlinear softmax, plus a skip connection, and normalization; (b) one layer of ResNet; (c) one layer of ResNeXt.

the data $\boldsymbol{X}$ as input, each layer's operators and parameters $(\boldsymbol{E}_\ell, \boldsymbol{C}_\ell)$ are constructed and initialized in an entirely *forward unrolling* fashion. This differs from the popular practice in deep learning: starting with a randomly constructed and initialized network which is then tuned globally via backward propagation (Rumelhart et al., 1986). It is widely believed that the brain is unlikely to use backward propagation as its learning mechanism due to the requirement for symmetric synapses and the complex form of feedback. Here, the forward unrolling optimization relies only on operations between adjacent layers that can be hard-wired; hence, it would be much easier for nature to realize and exploit.

Additionally, parameters and operators of the so-constructed networks are amenable to further fine-tuning via another level of optimization, e.g., (stochastic) gradient descent realized by backward propagation (Rumelhart et al., 1986)[14]. However, one should not confuse the (stochastic) gradient descent used to fine-tune a network with the gradient-based optimization that layers of the network ought to realize.

2.1.5 CNN derived from shift-invariance and nonlinearity

If we further wish the learned encoding f to be *invariant* (or equivariant) to all time-shifts or space-translations, then we view every sample $\boldsymbol{x}(t)$ with all its shifted versions $\{\boldsymbol{x}(t-\tau), \forall \tau\}$ as in the same equivalence class. If we compress and linearize them together into the same subspace, then all the linear operators, $\boldsymbol{E}$ or $\boldsymbol{C}$'s, in the above gradient operation (8) automatically become *multi-channel convolutions* (Chan KHR et al., 2022)! As a result, ReduNet naturally becomes a multi-channel convolutional neural network (CNN), originally proposed for shift-invariant recognition (Fukushima, 1980; LeCun et al., 1998)[15].

2.1.6 Artificial selection and evolution of neural networks

Once we realize that the role of the deep networks themselves is to conduct (gradient-based) iterative optimization to compress, linearize, and sparsify data, it may become easy to understand the "evolution" of artificial neural networks that has occurred in the past decade. Particularly, it helps explain why only a few have emerged on top through a process of *artificial selection*: going from general multi-layer perceptrons (MLPs) to CNNs to ResNets to Transformers. In comparison, a random search of network structures, such as neural architecture search (Baker et al., 2017; Zoph and Le, 2017) and AutoML (Hutter et al., 2019), has not resulted in any network architecture that is effective for general tasks. We speculate that successful architectures are

[14]It has been shown that ReduNets have the same model capacity (say to interpolate all training data precisely) as tried-and-tested deep networks such as ResNets (Chan KHR et al., 2022).

[15]In addition, due to special structures in such convolution operators $\boldsymbol{E}$ and $\boldsymbol{C}$'s, they are much more efficient to compute in the *frequency domain* than in the time/space domain: the computational complexity reduces from $O(D^3)$ (notice that computing $\boldsymbol{E}_\ell$ requires inverting a $D \times D$ matrix $\alpha(\boldsymbol{I} + \alpha \boldsymbol{Z}_\ell \boldsymbol{Z}_\ell^*)^{-1}$, which is in general of complexity $O(D^3)$) to $O(D)$ in the dimension D of the input signals (Chan KHR et al., 2022).

simply getting more and more effective and flexible at emulating iterative optimization schemes for data compression. Besides the aforementioned similarity between ReduNet and ResNet/ResNeXt, we want to discuss a few more examples.

2.1.7 What is a transformer transforming?

Notice that the gradient of a rate distortion term $R(\boldsymbol{Z})$ is of the form (7): $\frac{\partial R}{\partial \boldsymbol{Z}} = \alpha(\boldsymbol{I} + \alpha \boldsymbol{Z}_\ell \boldsymbol{Z}_\ell^*)^{-1}\boldsymbol{Z}_\ell$. Instead of viewing the matrix $\alpha(\boldsymbol{I} + \alpha \boldsymbol{Z}_\ell \boldsymbol{Z}_\ell^*)^{-1}$ as a linear operator $\boldsymbol{E}_\ell$ acting on $\boldsymbol{Z}_\ell$, as was done in ReduNet, we may rewrite the whole gradient term approximately as

$$\begin{aligned}\alpha(\boldsymbol{I} + \alpha \boldsymbol{Z}_\ell \boldsymbol{Z}_\ell^*)^{-1}\boldsymbol{Z}_\ell &\approx \alpha(\boldsymbol{I} - \alpha \boldsymbol{Z}_\ell \boldsymbol{Z}_\ell^*)\boldsymbol{Z}_\ell \\ &= \alpha\big[\boldsymbol{Z}_\ell - \alpha \boldsymbol{Z}_\ell(\boldsymbol{Z}_\ell^*\boldsymbol{Z}_\ell)\big].\end{aligned} \tag{9}$$

That is, the gradient operation for optimizing a rate distortion term depends mainly on the auto-correlation of the features $\boldsymbol{A} \doteq \boldsymbol{Z}_\ell^*\boldsymbol{Z}_\ell$ from the previous iteration. This is also known as "self-attention" or "self-expression" in some contexts (Vaswani et al., 2017; Vidal, 2022). If we consider applying an additional *learnable* linear transform $\boldsymbol{U}$ to each feature term in the above expression (9) for the gradient, a gradient-based iteration to optimize rate distortion takes the general form:

$$\boldsymbol{Z}_{\ell+1} \doteq \boldsymbol{Z}_\ell + \boldsymbol{U}_o\big[\boldsymbol{Z}_\ell - \alpha \boldsymbol{U}_v \boldsymbol{Z}_\ell(\boldsymbol{U}_k\boldsymbol{Z}_\ell)^*(\boldsymbol{U}_q\boldsymbol{Z}_\ell)\big]. \tag{10}$$

This is of exactly the same form as the basic operation of each layer for a Transformer (Vaswani et al., 2017), i.e., a self-attention (SA) head followed by a feed-forward residual MLP operation[16].

Moreover, very similar to ResNeXt versus ResNet, for tasks such as image classification, it is found empirically better to use *multiple*, say k, SA heads in parallel in each layer (Dosovitskiy et al., 2021). In the context of rate reduction, these SA heads may be naturally interpreted as gradient terms associated with the multiple rate distortion terms in the rate reduction $\Delta R(\boldsymbol{Z}) = R(\boldsymbol{Z}) - \big[R(\boldsymbol{Z}_1) + R(\boldsymbol{Z}_2) + \ldots + R(\boldsymbol{Z}_k)\big]/k$. The learned linear transforms $(\boldsymbol{U}_k, \boldsymbol{U}_q, \boldsymbol{U}_v)$ in each SA head can be interpreted as "matched filters" or "sparsifying dictionaries"[17] that select and transform token sets (on submanifolds) that belong to the same category (of signals or images). Hence, we conjecture that layers of Transformer (10) emulate a more general family of gradient-based iterative schemes that optimize the rate reduction of all input token sets (on multiple submanifolds) by clustering, compressing, and linearizing them altogether.

Furthermore, gradient ascent or descent is the most basic type of optimization scheme. Networks based on unrolling such schemes (e.g., ReduNet) might not be the most efficient yet. One could anticipate that more advanced optimization schemes, such as accelerated gradient descent methods (Wright and Ma, 2022), could lead to more efficient deep network architectures in the future. Architecture wise, these accelerated methods require the introduction of *skip connections* across multiple layers. This may help explain, from an optimization perspective, why additional skip connections have often been found to improve network efficiency in practice, e.g., in highway networks (Srivastava RK et al., 2015) or dense networks (Huang et al., 2017).

2.2 How to learn: the principle of Self-consistency

"*Everything should be made as simple as possible, but not any simpler.*"

— Albert Einstein

The principle of *Parsimony* alone does not ensure that a learned model will capture all important information in the data sensed about the external world. For example, mapping each class to a one-dimensional "one-hot" vector, by minimizing the cross entropy, may be viewed as a form of being parsimonious. It may learn a good classifier, but the features learned would collapse to a singleton, known as *neural collapse* (Papyan et al., 2020). The so-learned features would no longer contain enough information to regenerate the original data. Even if we consider the more general class of LDR models, the rate reduction objective alone does not automatically determine the correct dimension of the ambient feature space. If the feature space dimension is too

[16]If the term in the bracket (i.e., $\big[\boldsymbol{Z}_\ell - \alpha \boldsymbol{U}_v \boldsymbol{Z}_\ell(\boldsymbol{U}_k\boldsymbol{Z}_\ell)^* \cdot (\boldsymbol{U}_q\boldsymbol{Z}_\ell)\big]$) is interpreted as to emulate the gradient (9) of rate distortion, the linear operator $\boldsymbol{U}_o$ can then be viewed to emulate a certain regularized gradient-based method. For instance, it can be used to model the inverse of the Fisher information matrix in the natural gradient descent (Kakade, 2001).

[17]Interested readers may see Zhai et al. (2020) for more details about the topic of sparse dictionary learning.

low, the model learned will under-fit the data; if it is too high, the model might over-fit[18].

More generally, we take the view that perception is distinct from the performance of specific tasks, and the goal of perception is to learn *everything* predictable about what is sensed. In other words, the intelligent system should be able to *regenerate the distribution of the observed data from the compressed representation* to the point that itself cannot distinguish internally despite its best effort. This view distinguishes our framework from existing ones that are customized to a specific class of tasks. Representative of such is the *information bottleneck* framework (Tishby and Zaslavsky, 2015; Shwartz-Ziv and Tishby, 2017; Saxe et al., 2019), which explains how only information in the data related to its class label is extracted via the deep networks. To govern the process of learning a fully faithful representation[19], we introduce a second principle:

The principle of Self-consistency: *An autonomous intelligent system seeks a most self-consistent model for observations of the external world by minimizing the internal discrepancy between the observed and the regenerated.*

The principles of *Self-consistency* and *Parsimony* are highly complementary and should always be used together. The principle of *Self-consistency* alone does not ensure any gain in compression or efficiency. Mathematically and computationally, it is easy and even trivial to fit any training data with over-parameterized models[20] or to ensure consistency by establishing one-to-one mappings between domains with the same dimensions without learning intrinsic structures in the data distribution[21]. Only through compression can an intelligent system be compelled to discover intrinsic low-dimensional structures within the high-dimensional sensory data, and transform and represent them in the feature space in the most compact way for future use. Also, only through compression can we easily understand why over-parameterization, e.g., by feature lifting with hundreds of channels, as normally done in DNNs, will not lead to over-fitting if its sheer purpose is to compress in the higher-dimensional feature space: lifting helps reduce the nonlinearity in the data[22], rendering it easier to compress and linearize[23]. The role of subsequent layers is to perform compression (and linearization), and in general, the more the layers, the better it is compressed[24].

So far, we have established that a mechanism is needed to determine if the compressed representation contains *all* the information that is sensed. In the remainder of this section, we will first introduce a general architecture for achieving this, a *generative model*, which can regenerate a sample from its compressed representation. Then, a difficult problem arises: how to sensibly measure the discrepancy between the sensed sample and the regenerated sample? We argue that for an autonomous system, there is one and only one solution to this: measuring their discrepancies in the internal feature space. Finally, we argue that the compressive encoder and the generator must learn together through a zero-sum game. Through these deductions, we derive a universal framework for learning that we believe is inevitable.

2.2.1 Auto-encoding and its caveats with computability

To ensure that the learned feature mapping f and representation $\boldsymbol{z}$ have correctly captured low-dimensional structures in the data, one can check if the compressed feature $\boldsymbol{z}$ can reproduce the original data $\boldsymbol{x}$, by some generating map g, parameterized by

[18]The first expansive or contrastive term in the rate reduction might over-expand the features to fill the space, due to noises or other variations.

[19]Although in this section, for simplicity, we focus our discussions on modeling 2D imagery data, we will discuss the perception of the 3D world in Section 3.1, as well as argue why perception needs to integrate recognition, reconstruction, and regeneration.

[20]Having a photographic memory is not intelligence. It is the same as fitting all the data in the world with a Big Model.

[21]That is the case with many popular methods for learning generative models of data, such as normalizing flows (Kobyzev et al., 2021), CycleGAN (Zhu et al., 2017), and diffusion probabilistic models (Ho et al., 2020). Although so-learned models might be useful for applications such as image generation or style transfer, they neither identify low-dimensional structures in the data distributions nor produce compact linear structures in the learned representations.

[22]Say, as in the scattering transforms (Bruna and Mallat, 2013) or random filters (Chan TH et al., 2015; Chan KHR et al., 2022).

[23]As Lao Tzu famously said in *Tao Te Ching*: "*That which shrinks must first expand.*"

[24]This naturally explains a seemingly mysterious phenomenon about deep networks: the "double-descent" phenomenon suggests that a deep model's test error becomes smaller as it gets larger, after reaching its peak at a certain interpolation point (Belkin et al., 2019; Yang et al., 2020).

η:

$$\boldsymbol{x} \in \mathbb{R}^D \xrightarrow{f(\boldsymbol{x},\theta)} \boldsymbol{z} \in \mathbb{R}^d \xrightarrow{g(\boldsymbol{z},\eta)} \hat{\boldsymbol{x}} \in \mathbb{R}^D, \quad (11)$$

in the sense that $\hat{\boldsymbol{x}} = g(\boldsymbol{z}, \eta)$ is close to $\boldsymbol{x}$ (according to a certain measure). This process is generally known as *auto-encoding* (Kramer, 1991; Hinton and Zemel, 1993). In the special case of compressing to a *structured* representation such as LDR, we call such an auto-encoding a *transcription*[25] (Dai et al., 2022). However, this goal is easier said than done. The main difficulty lies in how to make this goal computationally tractable and hence physically realizable. More precisely, what is a principled measure for the difference between the distribution of $\boldsymbol{x}$ and that of $\hat{\boldsymbol{x}}$ that is both *mathematically well-defined* and *efficiently computable*? As mentioned before, when dealing with distributions in high-dimensional spaces with degenerate low-dimensional supports, which is almost always the case with real-world data (Ma et al., 2007; Vidal et al., 2016), conventional measures, including the KL divergence, mutual information, Jensen–Shannon distance, Helmholtz free energy, and Wasserstein distances, can be either ill-defined or intractable to compute, even for Gaussians (with support on subspaces) and their mixtures[26]. How can we resolve this fundamental and yet often unacknowledged difficulty in computability associated with comparing degenerate distributions in high-dimensional spaces?

2.2.2 Closed-loop data transcription for self-consistency

As shown in the previous Section 2.1, the rate reduction ΔR gives a well-defined principled distance measure between degenerate distributions. However, it is computable (with a closed form) only for a mixture of subspaces or Gaussians, not for general distributions! Yet, we can only expect the distribution of the internally structured representation $\boldsymbol{z}$ to be a mixture of subspaces or Gaussians, not the original data $\boldsymbol{x}$.

This leads to a rather profound question regarding learning a "self-consistent" representation: to verify the correctness of an internal model for the external world, *does an autonomous agent really need to measure any discrepancy in the data space*? The answer is actually no. The key is to realize that, to compare $\boldsymbol{x}$ and $\hat{\boldsymbol{x}} = g(\boldsymbol{z}, \eta)$, the agent needs only to compare their respective internal features $\boldsymbol{z} = f(\boldsymbol{x}, \theta)$ and $\hat{\boldsymbol{z}} = f(\hat{\boldsymbol{x}}, \theta)$ via the same mapping f that intends to make $\boldsymbol{z}$ compact and structured.

$$\boldsymbol{x} \xrightarrow{f(\boldsymbol{x},\theta)} \boldsymbol{z} \xrightarrow{g(\boldsymbol{z},\eta)} \hat{\boldsymbol{x}} \xrightarrow{f(\boldsymbol{x},\theta)} \hat{\boldsymbol{z}}. \quad (12)$$

Measuring distribution differences in the $\boldsymbol{z}$ space is well-defined and efficient: it is arguably true that in the case of natural intelligence, learning to measure discrepancies *internally* is the only thing that the brain of a self-contained autonomous agent can do[27].

This effectively leads to a "closed-loop" feedback system, and the overall process is illustrated in Fig. 6. The encoder f now plays an additional role as a discriminator, detecting any discrepancy between $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$ through the difference between their internal features $\boldsymbol{z}$ and $\hat{\boldsymbol{z}}$. The distance between the distribution of $\boldsymbol{z}$ and that of $\hat{\boldsymbol{z}}$ can be measured through the rate reduction (Eq. (2)) of their samples $\boldsymbol{Z}(\theta)$ and $\hat{\boldsymbol{Z}}(\theta, \eta)$:

$$\Delta R\big(\boldsymbol{Z}(\theta), \hat{\boldsymbol{Z}}(\theta,\eta)\big) \doteq R\big(\boldsymbol{Z} \cup \hat{\boldsymbol{Z}}\big) - \frac{1}{2}\big(R(\boldsymbol{Z}) + R(\hat{\boldsymbol{Z}})\big).$$

One can interpret popular practices for learning either a DNN classifier f or a generator g alone as learning an open-ended segment of the closed-loop system (Fig. 6). This currently popular practice is very similar to an open-loop control which has long been known in the control community to be problematic and costly. The training of such an open segment requires supervision on the desired output (e.g., class labels), and deployment of such an open-loop system is inherently not stable, robust, or adaptive if the data distributions, system parameters, or tasks change. For example, deep classification networks trained in supervised settings often suffer from *catastrophic forgetting* if retrained for new tasks with

[25] This is analogous to the memory-forming transcription process of engram (Josselyn and Tonegawa, 2020) or that between functional proteins and DNA (genes).

[26] Many existing methods formulate their objectives based on these quantities. Thus, these methods typically rely on expensive brute-force sampling to approximate these quantities or optimize their approximated lower-bounds or surrogates, such as in variational auto-encoding (VAE) (Kingma and Welling, 2013). The fundamental limitations of these methods are often disguised by good empirical results obtained using clever heuristics and excessive computational resources.

[27] Imagining someone colorblind, it is unlikely that his/her internal representation of the world requires minimizing discrepancies in RGB values of the visual inputs $\boldsymbol{x}$.

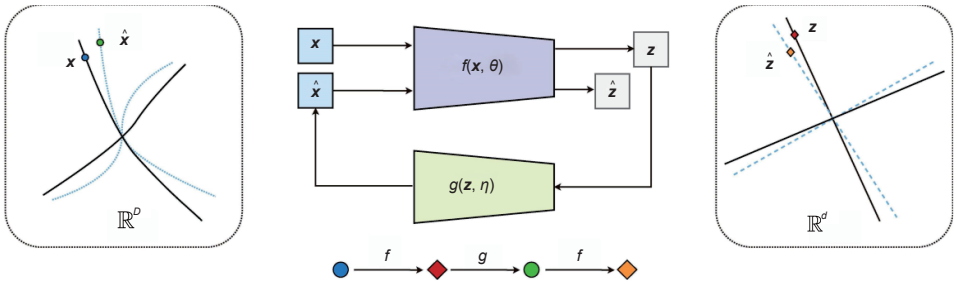

Fig. 6 A compressive closed-loop transcription of nonlinear data submanifolds to a linear discriminative representation (LDR), by comparing and minimizing the difference in $\boldsymbol{z}$ and $\hat{\boldsymbol{z}}$, internally. This leads to a natural pursuit-evasion game between the encoder/sensor f and the decoder/controller g, allowing the distribution of the decoded $\hat{\boldsymbol{x}}$ (the dotted blue curves) to chase and match that of the observed data $\boldsymbol{x}$ (the solid black curves).

new classes of data (McCloskey and Cohen, 1989). In contrast, closed-loop systems are inherently more stable and adaptive (Wiener, 1948). It has been suggested by Hinton et al. (1995) that the discriminative and generative segments need to be combined as the "wake" and the "sleep" phases, respectively, of a complete learning process.

2.2.3 Self-learning through a self-critiquing game

However, just closing the loop is not enough. It is tempting to think that now we need only to optimize the generator g to minimize the difference between $\boldsymbol{z}$ and $\hat{\boldsymbol{z}}$[28], e.g., in terms of the rate reduction measure:

$$\min_{\eta} \Delta R\big(\boldsymbol{Z}(\theta), \hat{\boldsymbol{Z}}(\theta,\eta)\big). \tag{13}$$

Note that $\Delta R(\boldsymbol{Z}, \hat{\boldsymbol{Z}}) = 0$ if $\hat{\boldsymbol{Z}} = g(f(\boldsymbol{Z})) = \boldsymbol{Z}$. That is, the optimal set of features $\boldsymbol{Z}$ should be a "fixed point" of the encoding-decoding loop[29]. However, the encoder f performs significant dimension reduction and compression, so $\hat{\boldsymbol{Z}} = \boldsymbol{Z}$ does not necessarily imply $\hat{\boldsymbol{X}} = \boldsymbol{X}$. To see this, consider the simplest case when $\boldsymbol{X}$ are already on a linear subspace (e.g., of dimension k) and f and g are a linear projection and lifting, respectively (Pai et al., 2022). f would not be able to detect any difference in its (large) null space: $\boldsymbol{X}$ and any $\hat{\boldsymbol{X}} = \boldsymbol{X} + \mathrm{null}(f)$ have the same image under f.

How can $\hat{\boldsymbol{Z}} = \boldsymbol{Z}$ imply $\hat{\boldsymbol{X}} = \boldsymbol{X}$ then? In other words, how can satisfaction with the self-consistency criterion in the internal space guarantee that we have learned to regenerate the observed data faithfully? This is possible *only when the dimension k is low enough and f can be further adjusted.* Let us assume that the dimension of $\boldsymbol{X}$ is $k < d/2$, where d is the dimension of the feature space. Then $\hat{\boldsymbol{X}} = g(f(\boldsymbol{X}))$ under a linear lifting g is a subspace of k-dimension. The union of the two subspaces of $\boldsymbol{X}$ and $\hat{\boldsymbol{X}}$ is of dimension at most $2k < d$. Hence, if there is a difference between these two subspaces and f can be an arbitrary projection, we have $f(\boldsymbol{X}) \neq f(\hat{\boldsymbol{X}})$; i.e., $\boldsymbol{X} \neq \hat{\boldsymbol{X}}$ implies $\boldsymbol{Z} \neq \hat{\boldsymbol{Z}}$.

Hence, after g minimizes the error ΔR in Eq. (13), f needs to actively adjust and detect, in its full capacity, if there is remaining discrepancy between $\boldsymbol{X}$ and $\hat{\boldsymbol{X}}$, e.g., by maximizing the same measure ΔR. The process can be repeated between the encoder f and the decoder g, resulting in a natural *pursuit and evasion game*, as illustrated in Fig. 6.

In the 1961 edition of his book *Cybernetics*, Wiener (1961) added a supplementary chapter discussing learning through playing games. The games he described were mostly about an intelligent agent against an opponent or the world (which we will discuss in Section 3). Here we advocate the need of an *internal* game-like mechanism for any intelligent agent to be able to conduct self-learning via self-critique! What abides by is the notion of (non-cooperative) games as a universally effective way of learning (von Neumann and Morgenstern, 1944;

[28]This is very similar in spirit to the "sleep" phase of the wake-sleep scheme proposed by Hinton et al. (1995): it essentially tries to ensure that the encoding (recognition) network f produces a response $\hat{\boldsymbol{z}}$ to the regenerated $\hat{\boldsymbol{x}} = g(\boldsymbol{z})$ *consistent* with its origin $\boldsymbol{z}$.

[29]This can be viewed as a generalization to the "deep equilibrium" models (Bai et al., 2019) or the "implicit deep learning" models (El Ghaoui et al., 2021). Both interpret deep learning as conducting fixed-point computation from a feedback control perspective.

Nash, 1951): applying the current model or strategy repeatedly against an adversarial critique, hence continuously improving the model or strategy based on feedback received through a closed loop!

Within such a framework, the encoder f assumes a dual role. In addition to learning a representation $\boldsymbol{z}$ for the data $\boldsymbol{x}$ by maximizing the rate reduction $\Delta R(\boldsymbol{Z})$ (as done in Section 2.1), it should serve as a feedback "sensor" that actively detects any discrepancy between the data $\boldsymbol{x}$ and the generated $\hat{\boldsymbol{x}}$. Also, the decoder g assumes a dual role: it is a "controller" that corrects any discrepancy between $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$ detected by f, as well as a decoder trying to minimize the overall coding rate $\Delta R(\hat{\boldsymbol{Z}})$ needed to achieve this goal (subject to a given precision).

Therefore, the optimal "parsimonious" and "self-consistent" representation tuple $(\boldsymbol{z}, f, g)$ can be interpreted as the *equilibrium point* of a zero-sum game between $f(\cdot,\theta)$ and $g(\cdot,\eta)$, over a combined rate reduction based utility on $\boldsymbol{Z}(\theta)$ and $\hat{\boldsymbol{Z}}(\theta,\eta)$:

$$\max_{\theta}\min_{\eta}\Delta R\big(\boldsymbol{Z}\big)+\Delta R\big(\hat{\boldsymbol{Z}}\big)+\Delta R\big(\boldsymbol{Z},\hat{\boldsymbol{Z}}\big). \tag{14}$$

A recent analysis has rigorously shown that, in the case when the input data $\boldsymbol{X}$ lie on multiple linear subspaces, the desired optimal representation for $\boldsymbol{Z}$ is indeed the Stackelberg equilibria (Fiez et al., 2019; Jin et al., 2020) of a sequential maximin game over a rate reduction objective similar to the above (Pai et al., 2022). It remains an open problem for the case when $\boldsymbol{X}$ are on multiple nonlinear submanifolds. Nevertheless, compelling empirical evidence indicates that solving this game indeed provides excellent auto-encoding for real-world visual datasets (Dai et al., 2022), and automatically determines a subspace with a proper dimension for each class. It does not seem to suffer from problems like *mode collapsing* in training conventional generative models, such as generative adversarial networks (GANs) (Srivastava A et al., 2017). The so-learned representation is simultaneously *discriminative and generative*.

2.2.4 Self-consistent incremental and unsupervised learning

So far, we have discussed mainly the two principles in the supervised setting. In fact, a primary advantage of our framework is that it is most natural and effective for *self-learning* via self-supervision and self-critique. Additionally, since the rate reduction has sought explicit (subspace-type) representations for the learned structures[30], this makes it easy for past knowledge to be preserved when learning new tasks/data, as a prior (memory) to be kept *self-consistent*.

For more clarity, let us examine how the closed-loop transcription framework above can be naturally extended to the case of *incremental learning*—that is, to learn to recognize one class of objects at a time instead of simultaneously learning many classes. While learning the representation $\boldsymbol{Z}_{\text{new}}$ for a new class, one needs only to add the cost to objective (14) and ensure that the representation $\boldsymbol{Z}_{\text{old}}$ learned before for old classes remains *self-consistent* (a fixed point) through the closed-loop transcription: $\boldsymbol{Z}_{\text{old}} \approx \hat{\boldsymbol{Z}}_{\text{old}} = f(g(\boldsymbol{Z}_{\text{old}}))$. In other words, the above maximin game (14) becomes a game with constraints:

$$\begin{aligned}&\max_{\theta}\min_{\eta}\ \Delta R(\boldsymbol{Z})+\Delta R\big(\hat{\boldsymbol{Z}}\big)+\Delta R(\boldsymbol{Z}_{\text{new}},\hat{\boldsymbol{Z}}_{\text{new}})\\&\qquad\text{subject to } \Delta R\big(\boldsymbol{Z}_{\text{old}},\hat{\boldsymbol{Z}}_{\text{old}}\big)=0.\end{aligned} \tag{15}$$

Such a constrained game makes learning an incremental and dynamic process, enabling the learned transcription to *adapt* to new incoming data continuously. This process is illustrated in Fig. 7.

Recent empirical studies (Tong et al., 2022) have shown that this leads to arguably the first self-contained neural system with a fixed capacity that can incrementally learn good LDR representations *without suffering from catastrophic forgetting* (McCloskey and Cohen, 1989). Forgetting, if any, is rather graceful with such a closed-loop system. Additionally, when images of an old class are provided again to the system for review, the learned representation can be further *consolidated*—a characteristic very similar to that of human memory. In some sense, such a constrained closed-loop formulation ensures that the visual memory formation can be *Bayesian* and *adaptive*—characteristics hypothesized to be desirable for the brain (Friston, 2009).

Note that this framework is fundamentally conceived to work in an entirely unsupervised setting. Thus, even though for pedagogical purposes we present the principles assuming that class

[30] instead of a "hidden" or "latent" representation learned using a purely generative method such as GAN (Goodfellow et al., 2014) where the features are distributed randomly in the feature space.

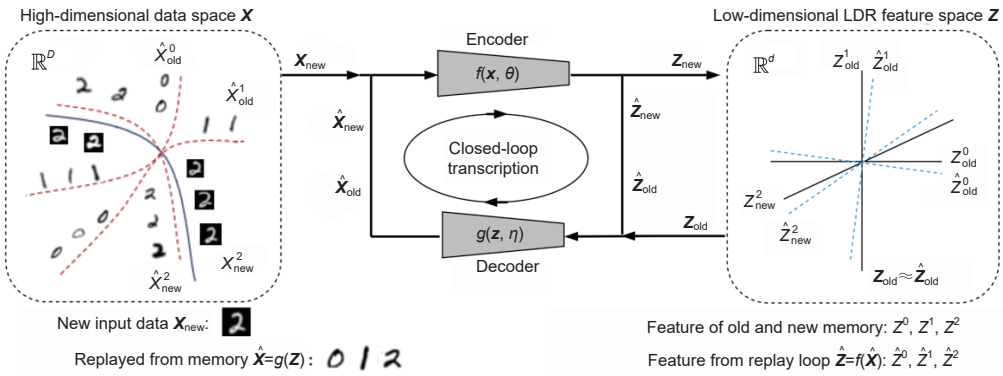

Fig. 7 Incremental learning via a compressive closed-loop transcription. For a new data class X_{new}, a new linear discriminative representation (LDR) memory Z_{new} is learned via a constrained minimax game between the encoder and decoder subject to a constraint that memory of past classes Z_{old} is preserved, as a "fixed point" of the closed loop.

(a) (b)

Fig. 8 Comparison between x and the corresponding decoded $\hat{x}$ of the auto-encoding learned in the unsupervised setting for the CIFAR-10 dataset (with 50 000 images in 10 classes) (a) and t-SNE of unsupervised-learned features of the 10 classes and visualization of several neighborhoods with their associated images (b). Notice the local thin (nearly 1D) structures in the visualized features, projected from a feature space of hundreds of dimensions.

information is available, the framework can be naturally extended to an entirely *unsupervised setting* in which no class information is given for any data sample. Here, we only have to view every new sample and its augmentations as one new class in Eq. (15). This can be viewed as one type of "self-supervision." With the "self-critiquing" game mechanism, a compressive closed-loop transcription can be easily learned. As shown in Fig. 8, the so-learned auto-encoding shows good sample-wise consistency, and the learned features also demonstrate clear and meaningful local low-dimensional (thin) structures. More surprisingly, subspaces or block-diagonal structures in the feature correlation emerge in the features learned for the classes even without any class information provided during training at all (Fig. 9)! Hence, structures of the so-learned features resemble those of category-selective areas observed in a primate's brain (Kanwisher et al., 1997; Kriegeskorte et al., 2008; Kanwisher, 2010; Bao et al., 2020).

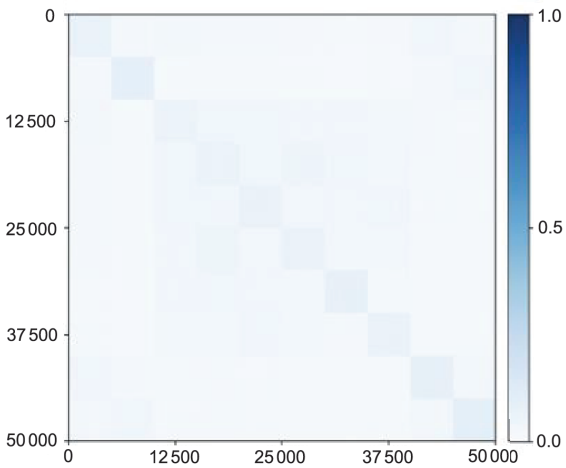

Fig. 9 Correlations between unsupervised-learned features for 50 000 images that belong to 10 classes (CIFAR-10) by the closed-loop transcription. Block-diagonal structures consistent with the classes emerge without any supervision.

3 Universal learning engines

"*What I cannot create, I do not understand.*"
— Richard Feynman

In the above section, we deduced from the first principles of *Parsimony* and *Self-consistency* the compressive closed-loop transcription framework, using the example of modeling visual imagery data. In the remaining two sections, we offer more speculative thoughts on the universality of this framework, extending it to 3D vision and reinforcement learning (RL) (the rest of this section)[31] and projecting its implications for neuroscience, mathematics, and higher-level intelligence (Section 4).

"Unite and build" versus "divide and conquer": Within the compressive closed-loop transcription framework, we have seen why and how fundamental ideas and concepts from coding/information theory, feedback control, deep networks, optimization, and game theory come together to become integral parts of a complete intelligent system that can learn. Although "divide and conquer" has long been a cherished tenet in scientific research, regarding understanding a complex system such as intelligence, the opposite "unite and build" should be the tenet of choice. Otherwise, we would forever be *blind men with an elephant*: each person would always believe that a small piece is the whole world and tend to blow its significance out of proportion[32].

The two principles serve as the glue needed to combine many necessary pieces together for the jigsaw puzzle of intelligence, with the role of deep networks naturally and clearly revealed as models for the nonlinear mappings between external observations and internal representations. Interestingly, the principles reveal computational mechanisms for learning systems that resemble some of the key characteristics observed in or hypothesized about the brain, such as sparse coding and subspace coding (Barlow, 1961; Olshausen and Field, 1996; Chang and Tsao, 2017), closed-loop feedback (Wiener, 1948), and free energy minimization (Friston, 2009), as we will discuss more in Section 4.

[31] Our discussions on the two topics require familiarity with certain domain-specific terminology and knowledge. Readers who are less familiar with these topics may skip without much loss of continuity.

[32] Hence all the superficial claims: "this or that is all you need."

Notice that closed-loop compressive architectures are ubiquitous for all intelligent beings and at all scales, from the brain (which compresses sensory information) to spinal circuits (which compress muscle movements) down to DNA (which compresses functional information about proteins). We believe that compressive closed-loop transcription may be the *universal learning engine* behind all intelligent behaviors. It enables intelligent beings and systems to discover and distill low-dimensional structures from seemingly complex and unorganized input and transform them into compact and organized internal structures for memorizing and exploitation.

To illustrate the universality of such a framework, for the remainder of this section, we examine two more tasks: *3D perception* and *decision making*, which are believed to be two key modules for any autonomous intelligent system (LeCun, 2022). We speculate on how, guided by the two principles, one can develop different perspectives and new insights to understand these challenging tasks.

3.1 Three-dimensional perception: closing the loop for vision and graphics

Thus far, we have demonstrated the success of closed-loop transcription in discovering compact structures in datasets of 2D images. This relies on the existence of *statistical correlations* among imagery data in each class. We believe that the same compression mechanisms would be even more effective if the low-dimensional structures in the data were defined through hard physical or geometric constraints rather than through soft statistical correlations.

Particularly, if we believe that the principles of *Parsimony* and *Self-consistency* also play a role in how the human brain develops mental models of the world from life-long visual inputs, then our sense of 3D space should be the result of such a closed-loop compression or transcription. The classic paradigm for 3D vision laid out by David Marr in his influential book *Vision* (Marr, 1982) advocates a "divide and conquer" approach that partitions the task of 3D perception into several modular processes: from low-level 2D processing (e.g., edge detection and contour sketching), to mid-level 2.5D parsing (e.g., grouping, segmentation, and figure and ground), and high-level 3D reconstruction (e.g., pose and shape) and recognition (e.g., objects). In contrast, the compressive

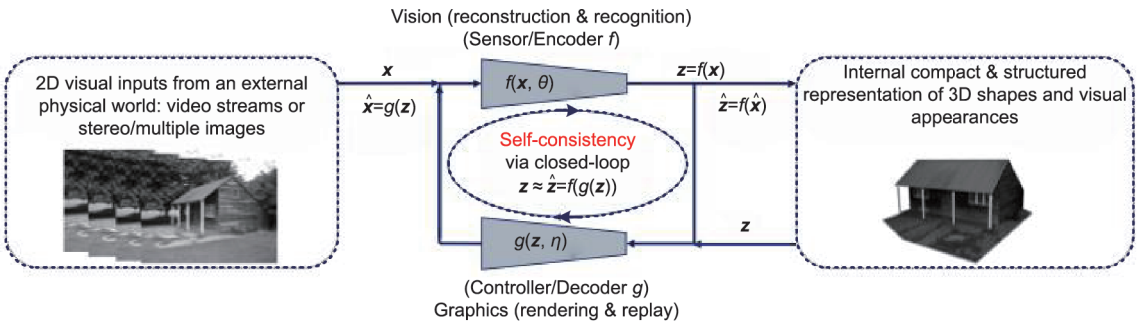

Fig. 10 A closed-loop relationship between computer vision and graphics for a compact and structured 3D model of the visual inputs.

closed-loop transcription proposed in this paper advocates an opposite "unite and build" approach.

3.1.1 Perception as a compressive closed-loop transcription?

More precisely, a 3D representation of shapes, appearances, and even dynamics of objects in the world should be the most compact and structured representation that our brain has developed internally to consistently interpret all perceived visual observations. If so, the two principles then suggest that a compact and structured 3D representation is directly the internal model to be sought for. This implies that we could and should unify computer vision and computer graphics within a single closed-loop computational framework, as illustrated in Fig. 10.

Computer vision has conventionally been interpreted as a forward process that reconstructs and recognizes an internal 3D model for the 2D visual inputs (Ma et al., 2004; Szeliski, 2022), whereas computer graphics (Hughes et al., 2014) represents its inverse process that renders and animates the internal 3D model. There might be tremendous computational and practical benefits to directly combine these two processes into a closed-loop system: all the rich structures (e.g., sparsity and smoothness) in geometric shapes, visual appearances, and dynamics can be exploited together for a unified 3D model that is the most compact and consistent with all visual inputs.

Indeed, the recognition techniques in computer vision could help computer graphics in building compact models in the spaces of shapes and appearance and enabling new ways for creating realistic 3D content. Conversely, the 3D modeling and simulation techniques in computer graphics could predict, learn, and verify the properties and behavior of the real objects and scenes analyzed by computer vision algorithms. In fact, the approach of "analysis by synthesis" has been long practiced by the vision and graphics community, e.g., for efficient online perception (Yildirim et al., 2020). Some recent examples of closing the loop for computer vision and graphics include a learned 3D rendering engine (Kulkarni et al., 2015) and 3D aware image synthesis (Chan ER et al., 2021; Wood et al., 2021).

3.1.2 Unified representations for appearance and shape?

Image-based rendering (Gortler et al., 1996; Levoy and Hanrahan, 1996; Shum et al., 2007), in which a new view is generated by learning from a set of given images, may be regarded as an early attempt to close the gap between vision and graphics with the principles of em Parsimony and *Self-consistency*. Particularly, plenoptic sampling (Chai et al., 2000) shows that an anti-aliased image (self-consistency) can be achieved with the minimum number of images required (parsimony).

Recent developments in modeling radiance fields have provided more empirical evidence for this view (Yu A et al., 2021): directly exploiting low-dimensional structures in the radiance field in 3D (sparse support and spatial smoothness) leads to much more efficient and effective solutions than brute-force training of black-box DNNs (Mildenhall et al., 2020). However, it remains a challenge for the future to identify the right family of compact and structured 3D representations that can integrate shape geometry, appearance, and even dynamics in a unified framework that leads to minimal complexity in data, model, and computation.

3.2 Decision making: closing the loop for perception, learning, and action

Thus far in this paper, we have discussed how compressive closed-loop transcription may lead to an effective and efficient framework for learning a good perceptual model from visual inputs. At the next level, an autonomous agent can use such a perceptual model to achieve certain tasks in a complex *dynamical* environment. The overall process for the agent to learn from perceived results or received rewards for its actions forms another closed loop at a higher level (Fig. 11).

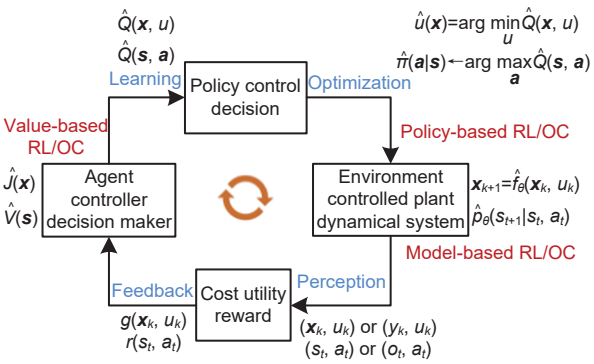

Fig. 11 An autonomous intelligent agent that integrates perception (feedback), learning, optimization, and action in a closed loop to learn an optimal policy for a certain task. s_t or x_k is the state of the world model; r or g is the perceived reward or cost of action a_t or control u_k on the current state; J or V is the (learned) cost or value associated with each state, Q is the (learned) cost associated with each state-action pair. Here, we deliberately use terminologies from optimal control (OC) (Bertsekas, 2012) and reinforcement learning (RL) (Sutton and Barto, 2018) in parallel for both comparison and unification.

The principle of *Self-consistency* is clearly at play here: the role of the closed-loop feedback system is to ensure that the learned model and control policy by the agent are *consistent* with the external world in such a way that the model can make the best prediction of the state (s_t) transition, and the learned control policy π_θ for the action (a_t) results in maximal expected reward R[33]:

$$\max_{\theta} R(\theta) \doteq \mathbb{E}_{a_t\sim\pi_\theta(s_t)}\Big[\sum_t r(s_t, a_t)\Big]. \tag{16}$$

[33]In many practices of RL, people may consider a "narrower" version of the self-consistency principle: it requires only the learned state model and control policy consistent with a specific task or reward, not a full state model for all sensed data.

Note here that the reward R plays a similar role as the rate reduction objective (4) for LDR models, which measures the "goodness" of the learned control policy π and guides its improvement.

The principle of *Parsimony* is the main reason for the success of modern RL in tackling large-scale tasks such as Alpha-Go (Silver et al., 2016, 2017) and playing video games (Berner et al., 2019; Vinyals et al., 2019). In almost all tasks that have a state-action space of astronomical size or dimension, e.g., D, practitioners always assume that the optimal value function V^*, Q-function Q^*, or policy π^* depends only on a small number of, e.g., $d \ll D$, features:

$$\begin{cases} V^*(\boldsymbol{s}) \approx \hat{V}\big(f(\boldsymbol{s}, \boldsymbol{a})\big), \\ Q^*(\boldsymbol{s}, \boldsymbol{a}) \approx \hat{Q}\big(f(\boldsymbol{s}, \boldsymbol{a})\big), \\ \pi^*(\boldsymbol{a} \mid \boldsymbol{s}) \approx \hat{\pi}\big(\boldsymbol{a}; f(\boldsymbol{s}, \boldsymbol{a})\big), \end{cases} \tag{17}$$

where $f(\boldsymbol{s}, \boldsymbol{a}) \in \mathbb{R}^d$ is a nonlinear mapping that learns some low-dimensional features of the extremely large or high-dimensional state-action space. In the case of video games, the state dimension D is easily in millions and yet the number of features d needed to learn a good policy is typically only a few dozen or hundred! Very often, these optimal control policies or value/reward functions sought in OC/RL are even assumed to be a linear superposition of these features (Ng and Russell, 2000; Kakade, 2001):

$$\boldsymbol{\omega}^{\mathrm{T}} f(\boldsymbol{s}, \boldsymbol{a}) = \omega_1 f_1(\boldsymbol{s}, \boldsymbol{a}) + \omega_2 f_2(\boldsymbol{s}, \boldsymbol{a}) + \ldots + \omega_d f_d(\boldsymbol{s}, \boldsymbol{a}). \tag{18}$$

That is, the nonlinear mapping f is assumed to also be able to *linearize* the dependency of the policy/value/reward functions on the learned features[34].

3.2.1 Autonomous feature selection via a game?

Notice that all these practices in RL are very similar in spirit to the learning objectives under the principle of *Parsimony* stated in Section 2.1. Effectively exploiting the low-dimensional structures is the (only) reason the learning can be so *scalable* with such a high-dimensional state-action space, and correctly identifying and linearizing such low-dimensional structures is the key for the so-learned

[34]It is a common practice in systems theory to linearize any nonlinear dynamics before controlling them, through either nonlinear mappings known as the Koopman operators (Koopman, 1931) or feedback linearization (Sastry, 1999).

control policy to be *generalizable*[35]. Nevertheless, a proper choice in the number of features d remains heuristically designed by the human in practice. That makes the overall RL not autonomous. We believe that, for a closed-loop learning system to automatically determine the right number of features associated with a reward/task, one must extend the RL formulation (16) to a certain max-imin game[36], in a similar spirit to those studied in Section 2.2 for achieving *Self-consistency* for visual modeling.

3.2.2 Data and computational efficiency of RL?

Recently, there have been many theoretical attempts to explain the empirically observed efficiency of RL in terms of sampling and computation complexity of Markov decision processes (MDPs). However, any theory based on unstructured generic MDPs and reward functions would not be able to provide pertinent explanations to such empirical successes. For example, some of the best known bounds on the sample complexity for RL remain linear in cardinality of the state space and action $O(|\mathcal{S}||\mathcal{A}|)$ (Li et al., 2020), which does not explain the empirically observed efficiency of RL in large-scale tasks (such as Alpha-Go and video games) where the state or action spaces are astronomical.

We believe that the efficiency of RL in tackling many practical large-scale tasks can come *only* from the intrinsic low dimensionality in the system dynamics or correlation between the optimal policy/control and the states. For example, assume that the systems have a bounded eluder dimension (Osband and van Roy, 2014) or the MDPs are of low rank (Agarwal et al., 2020; Uehara et al., 2022). Deep networks' role is again to identify and model such low-dimensional structures and hopefully linearize them.

To conclude, for large-scale RL tasks, the two principles together make such a closed-loop system of perception, learning, and action a truly efficient and effective learning engine. With such an engine, autonomous agents can discover low-dimensional structures if there are indeed such structures in the environment and the learning task, and eventually act intelligently when the structures learned are good enough and generalize well!

[35]Otherwise, the learned model/policy tends to over-fit or under-fit.

[36]by introducing a self-critique of the features selected and learned.

4 A broader program for intelligence

"*If I were to choose a patron saint for cybernetics out of the history of science, I should have to choose Leibniz. The philosophy of Leibniz centers about two closely related concepts—that of a universal symbolism and that of a calculus of reasoning.*"

— Norbert Wiener, *Cybernetics*, 1961

It has been 10 years since the dramatic revival of DNNs with the work of Krizhevsky et al. (2012), which has garnered considerable enthusiasm for artificial intelligence in both the technology industry and the scientific community. Subsequent theoretical studies of deep learning often view deep networks themselves as the object of study (Roberts and Yaida, 2022). However, we argue here that deep networks are better understood as a means to an end: they clearly arise to serve the purposes of identifying and transforming nonlinear low-dimensional structures in high-dimensional data, a universal task for learning from high-dimensional data (Wright and Ma, 2022).

More broadly, in this paper, we have proposed and argued that *Parsimony* and *Self-consistency* are two fundamental principles responsible for the emergence of intelligence, artificial or natural. The two principles together lead to a closed-loop computational framework that unifies and clarifies many practices and empirical findings of deep learning and artificial intelligence. Furthermore, we believe that they will guide us from now on to study intelligence with a more principled and integrated approach. Only in doing so can we achieve a new level of understanding of the science and mathematics behind intelligence.

4.1 Neuroscience of intelligence

One would naturally expect any fundamental principle of intelligence to have major implications for the design of the most intelligent thing in the universe, the brain. As already mentioned, the principles of *Parsimony* and *Self-consistency* shed new light on several experimental observations concerning the primate visual system. More importantly, they shine light on what to look for in future

experiments.

We have shown that seeking an internally parsimonious and predictive representation alone is enough "self-supervision" to allow structures to emerge automatically in the final representation learned through a compressive closed-loop transcription. For example, Fig. 9 shows that unsupervised data transcription learns features that automatically distinguish different categories, providing an explanation for category-selective representations observed in the brain (Kanwisher et al., 1997; Kriegeskorte et al., 2008; Kanwisher, 2010; Bao et al., 2020). These features also provide a plausible explanation for the widespread observations of sparse coding (Olshausen and Field, 1996) and subspace coding (Chang and Tsao, 2017; Bao et al., 2020) in the primate's brain. Furthermore, beyond the modeling of visual data, recent studies in neuroscience suggest that the emergence of other structured representations in the brain, such as "place cells," might also be the result of coding spatial information in the most compressed way (Benna and Fusi, 2021).

Arguably, the maximal coding rate reduction (MCR^2) principle (4) is similar in spirit to the "free energy minimization principle" from cognitive science (Friston, 2009), which attempts to provide a framework for Bayesian inference through energy minimization. Unlike the general notion of free energy, however, the rate reduction is computationally tractable and directly optimizable as it can be expressed in closed form. Furthermore, the interplay of our two principles suggests that autonomous learning of the correct model (class) should be conducted via a closed-loop maximin game over such a utility (14), instead of minimization alone. Thus, we believe that our framework provides a new perspective on how to practically implement Bayesian inference.

Our framework clarifies the overall learning architecture used by the brain. One important insight is that a feed-forward segment can be constructed by unrolling an optimization scheme; learning from a random network via back propagation is unnecessary. Furthermore, our framework suggests the existence of a complementary generative segment to form a closed-loop feedback system to guide learning. Finally, our framework sheds light on the elusive "prediction error" signal sought by many neuroscientists interested in brain mechanisms for "predictive coding," a computational scheme with resonances to compressive closed-loop transcription (Rao and Ballard, 1999; Keller and Mrsic-Flogel, 2018). For reasons of computational tractability, the discrepancy between incoming and generated observations should be measured at the final stage of representation.

So far, many resemblances between this new framework and the natural intelligence discussed in this paper are still speculative and remain to be corroborated with future scientific experiments and evidence. Nevertheless, these speculations offer neuroscientists ample new perspectives and hypotheses about why and how intelligence could emerge in nature.

4.2 Mathematics of intelligence

In terms of mathematical or statistical models for data analysis, one can view our framework as a generalization of principal component analysis (PCA) (Jolliffe, 1986), generalized PCA (GPCA) (Vidal et al., 2016), robust PCA (RPCA) (Candès et al., 2011), and nonlinear PCA (Kramer, 1991) to the case with multiple low-dimensional nonlinear submanifolds in a high-dimensional space. These classical methods largely model data with single or multiple linear subspaces or with a single nonlinear submanifold. We have argued that the role of deep networks is mainly to model the nonlinear mappings that simultaneously linearize and separate multiple low-dimensional submanifolds. This generalization is necessary to connect these idealistic, classic models to the true structures of the real-world data. Despite promising and exciting empirical evidence, the mathematical properties of the compressive closed-loop transcription process remain understudied and poorly understood. To the best of our knowledge, only for the case when the original data $\boldsymbol{x}$ are assumed to be on multiple linear subspaces, it has been shown that the maximin game based on rate reduction yields the correct optimal solution (Pai et al., 2022). Little is known about the transcription of nonlinear submanifolds.

A rigorous and systematic investigation requires an understanding of high-dimensional probability distributions with low-dimensional supports on submanifolds (Fefferman et al., 2013). Hence, mathematically, it is crucial to study how such submanifolds in high-dimensional spaces can be identified, grouped or separated, deformed, and flattened with

minimal distortion to their original metric, geometry, and topology (Tenenbaum et al., 2000; Buchanan et al., 2021; Wang et al., 2021; Shamir et al., 2022). Problems like these seem to fall into an understudied area between classical differential geometry and differential topology in mathematics.

Additionally, we often wish that during the process of deformation, the probability measure of data on each submanifold is redistributed in a certain optimal way such that coding and sampling will be the most economical and efficient. This is related to topics such as optimal transport (Lei et al., 2017). For the case when the submanifolds are fixed linear subspaces, understanding the achievable extremals of the rate reduction, or ratio of volumes of the whole and the parts, seems related to certain fundamental inequalities in analysis, such as the *Brascamp–Lieb* inequalities (Bennett et al., 2008). More general problems like these seem to be related to the studies of metric entropy (also commonly known as sphere packing) and coding theory for distributions over more general compact structures or spaces.

Besides nonlinear low-dimensional structures, real-world data and signals are typically *invariant* to shift in time, translation in space, or to more general group transformations. Wiener (1961) recognized that simultaneously dealing with nonlinearity and invariance presents a major technical challenge. He had made early attempts to generalize harmonic analysis to nonlinear processes and systems[37]. Indeed, the revival of deep learning has reignited strong interest in this critical problem, and significant progress has been made recently, including the work of Bruna and Mallat (2013), Cohen TS and Welling (2016), Papyan et al. (2018), Wiatowski and Bölcskei (2018), and Cohen TS et al. (2019). Our framework suggests that a more unifying approach to dealing with nonlinearity and invariance is through (incremental) compression. This has led to a natural derivation of structured deep networks such as the convolutional ReduNet (Chan KHR et al., 2022). We believe that compression provides a unifying perspective to modeling general *sequential* data or processes with nonlinear dynamics too, which could lead to mathematically rigorous justification for popular models such as recurrent neural networks (RNNs) or long short-term memory networks (LSTMs) (Hochreiter and Schmidhuber, 1997).

Besides pure mathematical interests, we must require that the mathematical investigation lead to *computationally tractable* measures and *scalable* algorithms. One must characterize the precise statistical and computational resources needed to achieve such tasks, in the same spirit as the research program set for compressive sensing (Wright and Ma, 2022), because intelligence needs to apply them to model high-dimensional data and solve large-scale tasks. This entails to "close the loop" between mathematics and computation, enabling the use of rich families of good geometric structures (e.g., sparse codes, subspaces, grids, groups, or graphs; Fig. 1, right) as compact archetypes for modeling real-world data, through efficiently computable nonlinear mappings that generalize deep networks, e.g., Bronstein et al. (2021).

4.3 Toward higher-level intelligence

The two principles laid out in this paper are mainly for explaining the emergence of intelligence in *individual* agents or systems, related to the notion of *ontogenetic learning* that Norbert Wiener first proposed (Wiener, 1948). It is probably noncoincidental that after more than 70 years, we find ourselves in this paper "closing the loop" of the modern practice of artificial intelligence back to its roots in *Cybernetics* and interweaving the very same set of fundamental concepts that Wiener touched upon in his book while conducting inquiries into the jigsaw puzzle of intelligence: *compact coding of information, closed-loop feedback, learning via games, white-box modeling, nonlinearity, shift-invariance, etc.*

As shown in this paper, the compressive closed-loop transcription is arguably the first computational framework that coherently integrates many of these pieces together. It is in close spirit to earlier frameworks (Hinton et al., 1995) but makes them computationally tractable and scalable. Particularly, the learned nonlinear encoding/decoding mappings of the loop, often manifested as deep networks, essentially provide a much needed "interface" between external unorganized raw sensory data (say visual, auditory, etc.) and internal compact and structured representations.

However, the two principles proposed in this paper do not necessarily explain all aspects of

[37] He used his analysis to explain brain waves (Wiener, 1961)!

intelligence. Computational mechanisms behind the emergence and development of high-level semantic, symbolic, or logic inferences remain elusive, although many foundational works have been set forth by pioneers like John McCarthy, Marvin Minsky, Allen Newell, and Herbert Simon since the 1950s (Simon, 1969; Newell and Simon, 1972) and a comprehensive modern exposition can be found in Russell and Norvig (2020). To date, there remain active and contentious debates about whether such high-level symbolic intelligence can emerge from continuous learning or must be hard-coded (Marcus, 2020; LeCun and Browning, 2022).

In our view, structured internal representations, such as subspaces, are necessary intermediate steps for the emergence of high-level semantic or symbolic concepts—each subspace corresponds to one *discrete* category (of objects). For example, notions of eye and mouth may come out naturally from observing a large number of face images. Additional statistical, causal, or logical relationships among the so-abstracted discrete concepts can be further modeled parsimoniously as a compact and structured (say sparse) graph, with each node representing a subspace/category, e.g., Bear et al. (2020). We believe that such a graph can be and should be learned via a closed-loop transcription to ensure self-consistency.

We conjecture that only on top of *compact and structured* representations learned by individual agents can the emergence and development of high-level intelligence (with shareable symbolic knowledge) be possible, subsequently and eventually. We suggest that new principles for the emergence of high-level intelligence, if any, should be sought through the need for efficient communication of information or transfer of knowledge among intelligent agents. This is related to the notion of *phylogenetic learning* that Wiener also discussed (Wiener, 1961). Furthermore, any new principle needed for such higher-level intelligence must reveal *reasons* why alignment and sharing of internal concepts across different individual agents is computationally possible, as well as reveal certain measurable *gains* in intelligence for a group of agents from such symbolic abstraction and sharing.

Intelligence as interpretable and computable systems: Obviously, as we advance our inquiries into higher-level intelligence, we want to set much higher standards this time. Whatever new principles that might remain to be discovered in the future, for them to truly play a substantial role in the emergence and development of intelligence, they must share two characteristics with the two principles we have presented in this study:

1. Interpretability: All principles together should help reveal computational mechanisms of intelligence as a white box[38], including measurable objectives, associated computing architectures, and structures of learned representations.

2. Computability: Any new principle for intelligence must be computationally tractable and scalable, physically realizable by computing machines or nature, and ultimately corroborated with scientific evidence.

Only with such fully interpretable and truly realizable principles in place can we explain all existing intelligent systems, artificial or natural, as partial or holistic instantiations of these principles. Then, they can help us discover effective architectures and systems for different intelligent tasks without relying on the current expensive and time-consuming "trial-and-error" approach to advance. Also, we will be able to characterize the minimal data and computation resources needed to achieve these tasks, instead of the current brute-force approach that advocates "the bigger, the better." Intelligence should not be the privilege of the most resourceful, as it is not the way of nature. Instead, parsimony and autonomy are the main characteristics[39]. Under a correct set of principles, anyone should be able to design and build future generations of intelligent systems, small or big, with autonomy, ability, and adaptiveness, eventually emulating and even surpassing that of animals and humans.

5 Conclusions

Through this paper, we hope to have convinced the reader that we are now at a much better place than people like Wiener and Shannon 70 years ago regarding uncovering, understanding, and exploiting the works of intelligence. We have proposed and

[38] Again, the phrase "white box" modeling has been conveniently borrowed from Wiener's *Cybernetics* (Wiener, 1961).

[39] A tiny ant is arguably much more intelligent and independent than any legged robot in the world, with merely a quarter of a million neurons consuming negligible energy.

argued that, under the two principles of *Parsimony* and *Self-consistency*, it is possible to assemble many necessary pieces of the puzzle of intelligence into a unified computational framework that is easily implementable on machines or by nature. This unifying framework offers new perspectives on how we could further advance the study of perception, decision making, and intelligence in general.

To conclude our proposal for a principled approach to intelligence, we emphasize once again that all scientific principles for intelligence should not be philosophical guidelines or conceptual frameworks formulated or developed with mathematical quantities that are intractable to compute or can only be approximated heuristically. They should rely on the most basic and principled objectives that are measurable with finite observations and lead to computational systems that can be realized even with limited resources. This belief is probably best expressed through a quote from Lord Kelvin[40]:

"*When you can measure what you are speaking about and express it in numbers, you know something about it; but when you cannot measure it, when you cannot express it in numbers, your knowledge is of the meager and unsatisfactory kind: it may be the beginning of knowledge, but you have scarcely, in your thoughts, advanced to the stage of science, whatever the matter may be.*"

— Lord Kelvin, 1883

Afterword and acknowledgements

Although the research of Yi and Harry focuses more on computer vision and computer graphics, they both happened to major in control and automation as undergraduate students. They started their collaboration many years ago at Microsoft Research Asia (MSRA) with a compression-based approach to data clustering and classification (Ma et al., 2007; Wright et al., 2007). In the past two years, they have had frequent discussions and debates about understanding (deep) learning and (artificial) intelligence. Their shared interests in intelligence have brought all these fundamental ideas together and led to the recent collaboration on closed-loop transcription (Dai et al., 2022), and eventually to many of the views shared in this paper. Doris is deeply interested in whether and how the brain implements generative models for visual perception, and her group has been having intense discussions with Yi on this topic since her moving to UC Berkeley a year ago.

The idea of writing this position paper is partly motivated by recent stimulating discussions among a group of researchers with very diverse backgrounds in artificial intelligence, applied mathematics, optimization, and neuroscience: Professors John Wright and Stefano Fusi of Columbia University, Professors Yann LeCun and Rob Fergus of New York University, Dr. Xin Tong of MSRA. We realize that these perspectives might be interesting to broader scientific and engineering communities.

Some of the thoughts presented about integrating pieces of the puzzle for intelligent systems can be traced back to an advanced robotics course that Yi had led and organized jointly with Professors Jitendra Malik, Shankar Sastry, and Claire Tomlin as Berkeley EECS290-005: *the Integration of Perception, Learning, and Control* in Spring 2021. The need for an integrated view or a "unite and build" approach seems to be a topic that is drawing increasing interest and importance for the study of artificial intelligence.

We would also like to thank many of our former and current students who, against extraordinary odds, have worked on projects under this new framework in the past several years when some of the ideas were still in their infancy and seemed not in accordance with the mainstream, including Xili Dai, Yaodong Yu, Peter Tong, Ryan Chan, Chong You, Ziyang Wu, Christina Baek, Druv Pai, Brent Yi, Michael Psenska, and others. Many of the technical evidence and figures used in this position paper are conveniently borrowed from their recent research results.

[40]that we have learned from Professor Jitendra Malik of UC Berkeley.

Contributors

The technical evidence of this paper is mainly based on research results from Yi MA's group in recent years, some of which are in collaboration with Heung-Yeung SHUM. Yi MA drafted the original manuscript. Heung-Yeung SHUM and Doris TSAO helped reorganize and revise the paper significantly. Particularly, Doris TSAO helped establish good connection of the proposed framework with studies and implications in neuroscience.

Compliance with ethics guidelines

Yi MA, Doris TSAO, and Heung-Yeung SHUM declare that they have no conflict of interest.

References

Agarwal A, Kakade S, Krishnamurthy A, et al., 2020. FLAMBE: structural complexity and representation

learning of low rank MDPs. Proc 34th Int Conf on Neural Information Processing Systems, p.20095-20107.

Azulay A, Weiss Y, 2019. Why do deep convolutional networks generalize so poorly to small image transformations? https://arxiv.org/abs/1805.12177

Baek C, Wu ZY, Chan KHR, et al., 2022. Efficient maximal coding rate reduction by variational forms. https://arxiv.org/abs/2204.00077

Bai SJ, Kolter JZ, Koltun V, 2019. Deep equilibrium models. Proc 33rd Int Conf on Neural Information Processing Systems, p.690-701.

Baker B, Gupta O, Naik N, et al., 2017. Designing neural network architectures using reinforcement learning. https://arxiv.org/abs/1611.02167

Bao PL, She L, McGill M, et al., 2020. A map of object space in primate inferotemporal cortex. *Nature*, 583(7814):103-108. https://doi.org/10.1038/s41586-020-2350-5

Barlow HB, 1961. Possible principles underlying the transformations of sensory messages. In: Rosenblith WA (Ed.), Sensory Communication. MIT Press, Cambridge, MA, USA, p.217-234.

Bear DM, Fan CF, Mrowca D, et al., 2020. Learning physical graph representations from visual scenes. Proc 34th Int Conf on Neural Information Processing Systems, p.6027-6039.

Belkin M, Hsu D, Ma SY, et al., 2019. Reconciling modern machine-learning practice and the classical bias-variance trade-off. *Proc Natl Acad Sci USA*, 116(32):15849-15854. https://doi.org/10.1073/pnas.1903070116

Benna MK, Fusi S, 2021. Place cells may simply be memory cells: memory compression leads to spatial tuning and history dependence. *Proc Natl Acad Sci USA*, 118(51):e2018422118. https://doi.org/10.1073/PNAS.2018422118

Bennett J, Carbery A, Christ M, et al., 2008. The Brascamp–Lieb inequalities: finiteness, structure and extremals. *Geom Funct Anal*, 17(5):1343-1415. https://doi.org/10.1007/s00039-007-0619-6

Berner C, Brockman G, Chan B, et al., 2019. Dota 2 with large scale deep reinforcement learning. https://arxiv.org/abs/1912.06680

Bertsekas DP, 2012. Dynamic Programming and Optimal Control, Volume I and II. Athena Scientific, Belmont, Massachusetts, USA.

Bronstein MM, Bruna J, Cohen T, et al., 2021. Geometric deep learning: grids, groups, graphs, geodesics, and gauges. https://arxiv.org/abs/2104.13478

Bruna J, Mallat S, 2013. Invariant scattering convolution networks. *IEEE Trans Patt Anal Mach Intell*, 35(8):1872-1886. https://doi.org/10.1109/TPAMI.2012.230

Buchanan S, Gilboa D, Wright J, 2021. Deep networks and the multiple manifold problem. https://arxiv.org/abs/2008.11245

Candès EJ, Li XD, Ma Y, et al., 2011. Robust principal component analysis? *J ACM*, 58(3):11. https://doi.org/10.1145/1970392.1970395

Chai JX, Tong X, Chan SC, et al., 2000. Plenoptic sampling. Proc 27th Annual Conf on Computer Graphics and Interactive Techniques, p.307-318. https://doi.org/10.1145/344779.344932

Chan ER, Monteiro M, Kellnhofer P, et al., 2021. pi-GAN: periodic implicit generative adversarial networks for 3D-aware image synthesis. https://arxiv.org/abs/2012.00926

Chan KHR, Yu YD, You C, et al., 2022. ReduNet: a white-box deep network from the principle of maximizing rate reduction. *J Mach Learn Res*, 23(114):1-103.

Chan TH, Jia K, Gao SH, et al., 2015. PCANet: a simple deep learning baseline for image classification? *IEEE Trans Image Process*, 24(12):5017-5032. https://doi.org/10.1109/TIP.2015.2475625

Chang L, Tsao DY, 2017. The code for facial identity in the primate brain. *Cell*, 169(6):1013-1028. https://doi.org/10.1016/j.cell.2017.05.011

Cohen H, Kumar A, Miller SD, et al., 2017. The sphere packing problem in dimension 24. *Ann Math*, 185(3):1017-1033. https://doi.org/10.4007/annals.2017.185.3.8

Cohen TS, Welling M, 2016. Group equivariant convolutional networks. https://arxiv.org/abs/1602.07576

Cohen TS, Geiger M, Weiler M, 2019. A general theory of equivariant CNNs on homogeneous spaces. Proc 33rd Int Conf on Neural Information Processing Systems, p.9145-9156.

Cover TM, Thomas JA, 2006. Elements of Information Theory (2nd Ed.). John Wiley & Sons, Inc., Hoboken, New Jersey, USA.

Dai XL, Tong SB, Li MY, et al., 2022. Closed-loop data transcription to an LDR via minimaxing rate reduction. https://arxiv.org/abs/2111.06636

Dosovitskiy A, Beyer L, Kolesnikov A, et al., 2021. An image is worth 16×16 words: transformers for image recognition at scale. https://arxiv.org/abs/2010.11929

El Ghaoui L, Gu FD, Travacca B, et al., 2021. Implicit deep learning. *SIAM J Math Data Sci*, 3(3):930-958. https://doi.org/10.1137/20M1358517

Engstrom L, Tran B, Tsipras D, et al., 2019. A rotation and a translation suffice: fooling CNNs with simple transformations. https://arxiv.org/abs/1712.02779v3

Fefferman C, Mitter S, Narayanan H, 2013. Testing the manifold hypothesis. https://arxiv.org/abs/1310.0425

Fiez T, Chasnov B, Ratliff LJ, 2019. Convergence of learning dynamics in Stackelberg games. https://arxiv.org/abs/1906.01217

Friston K, 2009. The free-energy principle: a rough guide to the brain? *Trends Cogn Sci*, 13(7):293-301. https://doi.org/10.1016/j.tics.2009.04.005

Fukushima K, 1980. Neocognitron: a self organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biol Cybern*, 36(4):193-202. https://doi.org/10.1007/BF00344251

Goodfellow IJ, Pouget-Abadie J, Mirza M, et al., 2014. Generative adversarial nets. Proc 27th Int Conf on Neural Information Processing Systems, p.2672-2680.

Gortler SJ, Grzeszczuk R, Szeliski R, et al., 1996. The lumigraph. Proc 23rd Annual Conf on Computer Graphics and Interactive Techniques, p.43-54. https://doi.org/10.1145/237170.237200

Gregor K, LeCun Y, 2010. Learning fast approximations of sparse coding. Proc 27th Int Conf on Machine Learning, p.399-406.

Hadsell R, Chopra S, LeCun Y, 2006. Dimensionality reduction by learning an invariant mapping. IEEE Computer Society Conf on Computer Vision and Pattern Recognition, p.1735-1742. https://doi.org/10.1109/CVPR.2006.100

He KM, Zhang XY, Ren SQ, et al., 2016. Deep residual learning for image recognition. IEEE Conf on Computer Vision and Pattern Recognition, p.770-778. https://doi.org/10.1109/CVPR.2016.90

Hinton GE, Zemel RS, 1993. Autoencoders, minimum description length and Helmholtz free energy. Proc 6th Int Conf on Neural Information Processing Systems, p.3-10.

Hinton GE, Dayan P, Frey BJ, et al., 1995. The "wake-sleep" algorithm for unsupervised neural networks. *Science*, 268(5214):1158-1161. https://doi.org/10.1126/science.7761831

Ho J, Jain A, Abbeel P, 2020. Denoising diffusion probabilistic models. https://arxiv.org/abs/2006.11239

Hochreiter S, Schmidhuber J, 1997. Long short-term memory. *Neur Comput*, 9(8):1735-1780. https://doi.org/10.1162/neco.1997.9.8.1735

Huang G, Liu Z, van der Maaten L, et al., 2017. Densely connected convolutional networks. IEEE Conf on Computer Vision and Pattern Recognition, p.2261-2269. https://doi.org/10.1109/CVPR.2017.243

Hughes JF, van Dam A, McGuire M, et al., 2014. Computer Graphics: Principles and Practice (3rd Ed.). Addison-Wesley, Upper Saddle River, NJ, USA.

Hutter F, Kotthoff L, Vanschoren J, 2019. Automated Machine Learning: Methods, Systems, Challenges. Springer Cham. https://doi.org/10.1007/978-3-030-05318-5

Hyvärinen A, 1997. A family of fixed-point algorithms for independent component analysis. IEEE Int Conf on Acoustics, Speech, and Signal Processing, p.3917-3920. https://doi.org/10.1109/ICASSP.1997.604766

Hyvärinen A, Oja E, 1997. A fast fixed-point algorithm for independent component analysis. *Neur Comput*, 9(7): 1483-1492. https://doi.org/10.1162/neco.1997.9.7.1483

Jin C, Netrapalli P, Jordan MI, 2020. What is local optimality in nonconvex-nonconcave minimax optimization? https://arxiv.org/abs/1902.00618

Jolliffe IT, 1986. Principal Component Analysis. Springer-Verlag, New York, NY, USA. https://doi.org/10.1007/978-1-4757-1904-8

Josselyn SA, Tonegawa S, 2020. Memory engrams: recalling the past and imagining the future. *Science*, 367(6473):eaaw4325. https://doi.org/10.1126/science.aaw4325

Kakade SM, 2001. A natural policy gradient. Proc 14th Int Conf on Neural Information Processing Systems: Natural and Synthetic, p.1531-1538.

Kanwisher N, 2010. Functional specificity in the human brain: a window into the functional architecture of the mind. *Proc Natl Acad Sci USA*, 107(25):11163-11170. https://doi.org/10.1073/pnas.1005062107

Kanwisher N, McDermott J, Chun MM, 1997. The fusiform face area: a module in human extrastriate cortex specialized for face perception. *J Neurosci*, 17(11): 4302-4311. https://doi.org/10.1523/JNEUROSCI.17-11-04302.1997

Keller GB, Mrsic-Flogel TD, 2018. Predictive processing: a canonical cortical computation. *Neuron*, 100(2):424-435. https://doi.org/10.1016/j.neuron.2018.10.003

Kelley HJ, 1960. Gradient theory of optimal flight paths. *ARS J*, 30(10):947-954. https://doi.org/10.2514/8.5282

Kingma DP, Welling M, 2013. Auto-encoding variational Bayes. https://arxiv.org/abs/1312.6114

Kobyzev I, Prince SJD, Brubaker MA, 2021. Normalizing flows: an introduction and review of current methods. *IEEE Trans Patt Anal Mach Intell*, 43(11):3964-3979. https://doi.org/10.1109/tpami.2020.2992934

Koopman BO, 1931. Hamiltonian systems and transformation in Hilbert space. *Proc Natl Acad Sci USA*, 17(5): 315-318. https://doi.org/10.1073/pnas.17.5.315

Kramer MA, 1991. Nonlinear principal component analysis using autoassociative neural networks. *AIChE J*, 37(2): 233-243. https://doi.org/10.1002/aic.690370209

Kriegeskorte N, Mur M, Ruff DA, et al., 2008. Matching categorical object representations in inferior temporal cortex of man and monkey. *Neuron*, 60(6):1126-1141. https://doi.org/10.1016/j.neuron.2008.10.043

Krizhevsky A, Sutskever I, Hinton GE, 2012. ImageNet classification with deep convolutional neural networks. Proc 25th Int Conf on Neural Information Processing Systems, p.1097-1105.

Kulkarni TD, Whitney WF, Kohli P, et al., 2015. Deep convolutional inverse graphics network. Proc 28th Int Conf on Neural Information Processing Systems, p.2539-2547.

LeCun Y, 2022. A Path Towards Autonomous Machine Intelligence. https://openreview.net/pdf?id=BZ5a1r-kVsf

LeCun Y, Browning J, 2022. What AI can tell us about intelligence. NO-EMA Magazine. https://www.noemamag.com/what-ai-can-tell-us-about-intelligence/

LeCun Y, Bottou L, Bengio Y, et al., 1998. Gradient-based learning applied to document recognition. *Proc IEEE*, 86(11):2278-2324. https://doi.org/10.1109/5.726791

LeCun Y, Bengio Y, Hinton G, 2015. Deep learning. *Nature*, 521(7553):436-444. https://doi.org/10.1038/nature14539

Lei N, Su KH, Cui L, et al., 2017. A geometric view of optimal transportation and generative model. https://arxiv.org/abs/1710.05488

Levoy M, Hanrahan P, 1996. Light field rendering. Proc 23rd Annual Conf on Computer Graphics and Interactive Techniques, p.31-42. https://doi.org/10.1145/237170.237199

Li G, Wei YT, Chi YJ, et al., 2020. Breaking the sample size barrier in model-based reinforcement learning with a generative model. Proc 34th Int Conf on Neural Information Processing Systems, p.12861-12872.

Ma Y, Soatto S, Košecká J, et al., 2004. An Invitation to 3-D Vision: from Images to Geometric Models. Springer-Verlag, New York, USA. https://doi.org/10.1007/978-0-387-21779-6

Ma Y, Derksen H, Hong W, et al., 2007. Segmentation of multivariate mixed data via lossy data coding and compression. *IEEE Trans Patt Anal Mach Intell*, 29(9): 1546-1562. https://doi.org/10.1109/TPAMI.2007.1085

MacDonald J, Wäldchen S, Hauch S, et al., 2019. A rate-distortion framework for explaining neural network decisions. https://arxiv.org/abs/1905.11092

Marcus G, 2020. The next decade in AI: four steps towards robust artificial intelligence. https://arxiv.org/abs/2002.06177

Marr D, 1982. Vision. MIT Press, Cambridge, MA, USA.

Mayr O, 1970. The Origins of Feedback Control. MIT Press, Cambridge, MA, USA.

McCloskey M, Cohen NJ, 1989. Catastrophic interference in connectionist networks: the sequential learning problem. *Psychol Learn Motiv*, 24:109-165. https://doi.org/10.1016/S0079-7421(08)60536-8

Mildenhall B, Srinivasan PP, Tancik M, et al., 2020. NeRF: representing scenes as neural radiance fields for view synthesis. https://arxiv.org/abs/2003.08934

Nash J, 1951. Non-cooperative games. *Ann Math*, 54(2):286-295. https://doi.org/10.2307/1969529

Newell A, Simon HA, 1972. Human Problem Solving. Prentice Hall, Englewood Cliffs, New Jersey, USA.

Ng AY, Russell SJ, 2000. Algorithms for inverse reinforcement learning. Proc 17th Int Conf on Machine Learning, p.663-670.

Olshausen BA, Field DJ, 1996. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607-609. https://doi.org/10.1038/381607a0

Osband I, van Roy B, 2014. Model-based reinforcement learning and the eluder dimension. Proc 27th Int Conf on Neural Information Processing Systems, p.1466-1474.

Pai D, Psenka M, Chiu CY, et al., 2022. Pursuit of a discriminative representation for multiple subspaces via sequential games. https://arxiv.org/abs/2206.09120

Papyan V, Romano Y, Sulam J, et al., 2018. Theoretical foundations of deep learning via sparse representations: a multilayer sparse model and its connection to convolutional neural networks. *IEEE Signal Process Mag*, 35(4):72-89. https://doi.org/10.1109/MSP.2018.2820224

Papyan V, Han XY, Donoho DL, 2020. Prevalence of neural collapse during the terminal phase of deep learning training. https://arxiv.org/abs/2008.08186

Patterson D, Gonzalez J, Hölzle U, et al., 2022. The carbon footprint of machine learning training will plateau, then shrink. https://arxiv.org/abs/2204.05149

Quinlan JR, 1986. Induction of decision trees. *Mach Learn*, 1(1):81-106. https://doi.org/10.1007/BF00116251

Rao RPN, Ballard DH, 1999. Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nat Neurosci*, 2(1):79-87. https://doi.org/10.1038/4580

Rifai S, Vincent P, Muller X, et al., 2011. Contractive auto-encoders: explicit invariance during feature extraction. Proc 28th Int Conf on Machine Learning, p.833-840.

Rissanen J, 1989. Stochastic Complexity in Statistical Inquiry. World Scientific Publishing Co., Inc., Singapore.

Roberts DA, Yaida S, 2022. The Principles of Deep Learning Theory. Cambridge University Press, Cambridge, MA, USA.

Rosenblatt F, 1958. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychol Rev*, 65(6):386-408. https://doi.org/10.1037/h0042519

Rumelhart DE, Hinton GE, Williams RJ, 1986. Learning representations by back-propagating errors. *Nature*, 323(6088):533-536. https://doi.org/10.1038/323533a0

Russell S, Norvig P, 2020. Artificial Intelligence: a Modern Approach (4th Ed.). Pearson Education, Inc., River Street, Hoboken, NJ, USA.

Sastry S, 1999. Nonlinear Systems: Analysis, Stability, and Control. Springer, New York, USA.

Saxe AM, Bansal Y, Dapello J, et al., 2019. On the information bottleneck theory of deep learning. *J Stat Mech*, 2019:124020. https://doi.org/10.1088/1742-5468/ab3985

Shamir A, Melamed O, BenShmuel O, 2022. The dimpled manifold model of adversarial examples in machine learning. https://arxiv.org/abs/2106.10151

Shannon CE, 1948. A mathematical theory of communication. *Bell Syst Techn J*, 27(3):379-423. https://doi.org/10.1002/j.1538-7305.1948.tb01338.x

Shazeer N, Mirhoseini A, Maziarz K, et al., 2017. Outrageously large neural networks: the sparsely-gated mixture-of-experts layer. https://arxiv.org/abs/1701.06538

Shum HY, Chan SC, Kang SB, 2007. Image-Based Rendering. Springer, New York, USA.

Shwartz-Ziv R, Tishby N, 2017. Opening the black box of deep neural networks via information. https://arxiv.org/abs/1703.00810

Silver D, Huang A, Maddison CJ, et al., 2016. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484-489. https://doi.org/10.1038/nature16961

Silver D, Schrittwieser J, Simonyan K, et al., 2017. Mastering the game of Go without human knowledge. *Nature*, 550(7676):354-359. https://doi.org/10.1038/nature24270

Simon HA, 1969. The Sciences of the Artificial. MIT Press, Cambridge, MA, USA.

Srivastava A, Valkoz L, Russell C, et al., 2017. VeeGAN: reducing mode collapse in GANs using implicit variational learning. Proc 31st Int Conf on Neural Information Processing Systems, p.3310-3320.

Srivastava RK, Greff K, Schmidhuber J, 2015. Highway networks. https://arxiv.org/abs/1505.00387

Sutton RS, Barto AG, 2018. Reinforcement Learning: an Introduction (2nd Ed.). MIT Press, Cambridge, MA, USA.

Szegedy C, Zaremba W, Sutskever I, et al., 2014. Intriguing properties of neural networks. https://arxiv.org/abs/1312.6199

Szeliski R, 2022. Computer Vision: Algorithms and Applications (2nd Ed.). Springer-Verlag, Switzerland. https://doi.org/10.1007/978-3-030-34372-9

Tenenbaum JB, de Silva V, Langford JC, 2000. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319-2323. https://doi.org/10.1126/science.290.5500.2319

Tishby N, Zaslavsky N, 2015. Deep learning and the information bottleneck principle. IEEE Information Theory Workshop, p.1-5. https://doi.org/10.1109/ITW.2015.7133169

Tong SB, Dai XL, Wu ZY, et al., 2022. Incremental learning of structured memory via closed-loop transcription. https://arxiv.org/abs/2202.05411

Uehara M, Zhang XZ, Sun W, 2022. Representation learning for online and offline RL in low-rank MDPs. https://arxiv.org/abs/2110.04652v1

van den Oord A, Li YZ, Vinyals O, 2019. Representation learning with contrastive predictive coding. https://arxiv.org/abs/1807.03748v1

Vaswani A, Shazeer N, Parmar N, et al., 2017. Attention is all you need. https://arxiv.org/abs/1706.03762

Viazovska MS, 2017. The sphere packing problem in dimension 8. *Ann Math*, 185(3):991-1015. https://doi.org/10.4007/annals.2017.185.3.7

Vidal R, 2022. Attention: Self-Expression Is All You Need. https://openreview.net/forum?id=MmujBClawFo

Vidal R, Ma Y, Sastry SS, 2016. Generalized Principal Component Analysis. Springer Verlag, New York, USA. https://doi.org/10.1007/978-0-387-87811-9

Vinyals O, Babuschkin I, Czarnecki WM, et al., 2019. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature*, 575(7782):350-354. https://doi.org/10.1038/s41586-019-1724-z

von Neumann J, Morgenstern O, 1944. Theory of Games and Economic Behavior. Princeton University Press, Princeton, NJ, USA.

Wang TR, Buchanan S, Gilboa D, et al., 2021. Deep networks provably classify data on curves. https://arxiv.org/abs/2107.14324

Wiatowski T, Bölcskei H, 2018. A mathematical theory of deep convolutional neural networks for feature extraction. *IEEE Trans Inform Theory*, 64(3):1845-1866. https://doi.org/10.1109/TIT.2017.2776228

Wiener N, 1948. Cybernetics. MIT Press, Cambridge, MA, USA.

Wiener N, 1961. Cybernetics (2nd Ed.). MIT Press, Cambridge, MA, USA.

Wisdom S, Powers T, Pitton J, et al., 2017. Building recurrent networks by unfolding iterative thresholding for sequential sparse recovery. IEEE Int Conf on Acoustics, Speech and Signal Processing, p.4346-4350. https://doi.org/10.1109/ICASSP.2017.7952977

Wood E, Baltrušaitis T, Hewitt C, et al., 2021. Fake it till you make it: face analysis in the wild using synthetic data alone. IEEE/CVF Int Conf on Computer Vision, p.3661-3671. https://doi.org/10.1109/ICCV48922.2021.00366

Wright J, Ma Y, 2022. High-Dimensional Data Analysis with Low-Dimensional Models: Principles, Computation, and Applications. Cambridge University Press, Cambridge, MA, USA. https://doi.org/10.1017/9781108779302

Wright J, Tao Y, Lin ZY, et al., 2007. Classification via minimum incremental coding length (MICL). Proc 20th Int Conf on Neural Information Processing Systems, p.1633-1640.

Xie SN, Girshick R, Dollár P, et al., 2017. Aggregated residual transformations for deep neural networks. IEEE Conf on Computer Vision and Pattern Recognition, p.5987-5995. https://doi.org/10.1109/CVPR.2017.634

Yang ZT, Yu YD, You C, et al., 2020. Rethinking bias-variance trade-off for generalization of neural networks. Proc 37th Int Conf on Machine Learning, p.10767-10777.

Yildirim I, Belledonne M, Freiwald W, et al., 2020. Efficient inverse graphics in biological face processing. *Sci Adv*, 6(10):eaax5979. https://doi.org/10.1126/sciadv.aax5979

Yu A, Fridovich-Keil S, Tancik M, et al., 2021. Plenoxels: radiance fields without neural networks. https://arxiv.org/abs/2112.05131

Yu YD, Chan KHR, You C, et al., 2020. Learning diverse and discriminative representations via the principle of maximal coding rate reduction. Proc 34th Int Conf on Neural Information Processing Systems, p.9422-9434.

Zeiler MD, Fergus R, 2014. Visualizing and understanding convolutional networks. Proc 13th European Conf on Computer Vision, p.818-833. https://doi.org/10.1007/978-3-319-10590-1_53

Zhai YX, Yang ZT, Liao ZY, et al., 2020. Complete dictionary learning via ℓ^4-norm maximization over the orthogonal group. *J Mach Learn Res*, 21(1):6622-6689.

Zhu JY, Park T, Isola P, et al., 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. IEEE Int Conf on Computer Vision, p.2242-2251. https://doi.org/10.1109/ICCV.2017.244

Zoph B, Le QV, 2017. Neural architecture search with reinforcement learning. https://arxiv.org/abs/1611.01578