

Lecture 8: Consistent and Self-Consistent Representations via Compressive Autoencoding and Transcription

Professor Yi Ma

School of Computing and Data Science
The University of Hong Kong

September 21, 2025

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

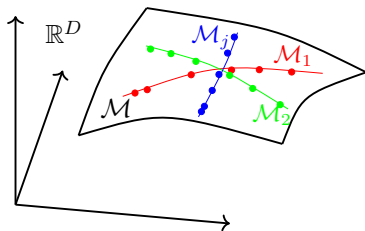
– Albert Einstein

Outline

- 1 Objective of Learning Representations for High-Dimensional Data
- 2 **Consistency**: Compressive Autoencoding
- 3 **Self-Consistency**: Closed-Loop Transcription
- 4 Empirical Verification
- 5 Incremental and Continuous Learning
- 6 Conclusions, Extensions, and Open Problems

Objective of Learning from High-Dimensional Data

Figure: High-dimensional Real-World Data: data samples $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_m]$ in \mathbb{R}^D lying on a mixture of low-dimensional submanifolds $\mathbf{X} \subset \cup_{j=1}^k \mathcal{M}_j \subset \mathbb{R}^D$.



Main objective of learning from sensed (or sampled) data of the real-world:

Seek a most compact and structured representation of the data.

A Special Case: Fitting Class Labels via a Deep Network

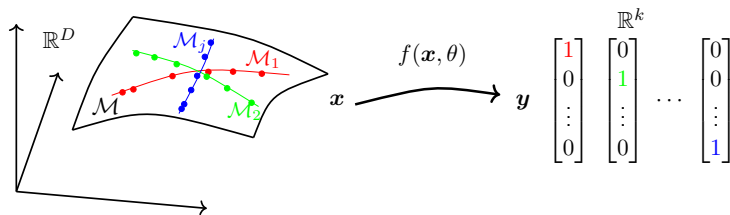
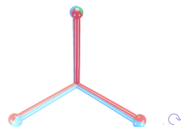


Figure: Black Box DNN for Classification: y is the class label of x represented as a “one-hot” vector in \mathbb{R}^k . To learn a nonlinear mapping $f(\cdot, \theta) : x \mapsto y$, say modeled by a deep network, using cross-entropy (CE) loss.

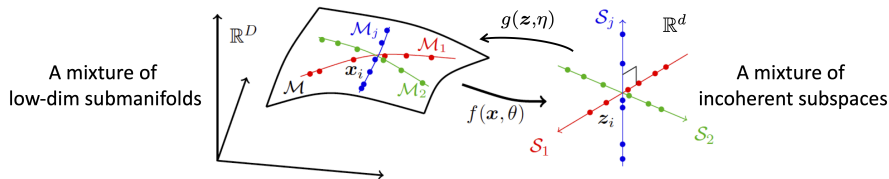
$$\min_{\theta \in \Theta} \text{CE}(\theta, x, y) \doteq -\mathbb{E}[\langle y, \log[f(x, \theta)] \rangle] \approx -\frac{1}{m} \sum_{i=1}^m \langle y_i, \log[f(x_i, \theta)] \rangle. \quad (1)$$

*Prevalence of **neural collapse** during the terminal phase of deep learning training, Papayan, Han, and Donoho, 2020.*



Parsimony: What to Learn from High-Dim Data?

Assumption: the data $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_m] \in \mathbb{R}^D$ lie on one or multiple low-dim submanifolds: $\mathbf{X} \subset \cup_{j=1}^k \mathcal{M}_j$ in a high-dim space $\in \mathbb{R}^D$:



Goal: learn a **linear discriminative representation** $\mathbf{Z} = [\mathbf{z}_1, \dots, \mathbf{z}_m] \in \mathbb{R}^d$ ($d \ll D$) for the data $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_m] \in \mathbb{R}^D$ such that \mathbf{Z} is the most informative:

$$\mathbf{X} \subset \mathbb{R}^D \xrightarrow{f(\mathbf{x}, \theta)} \mathbf{Z} \subset \mathbb{R}^d. \quad (2)$$

Parsimony: Measured by Rate Reduction/Information Gain

Difference in rate distortion between the whole and the parts:

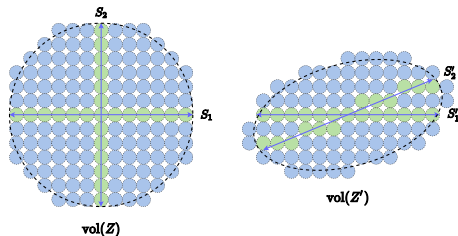
$$\Delta R(\mathbf{Z}, \mathbf{\Pi}, \epsilon) = \underbrace{\frac{1}{2} \log \det \left(\mathbf{I} + \frac{d}{m\epsilon^2} \mathbf{Z} \mathbf{Z}^\top \right)}_{R(\mathbf{Z})} - \underbrace{\sum_{j=1}^k \frac{\text{tr}(\mathbf{\Pi}_j)}{2m} \log \det \left(\mathbf{I} + \frac{d}{\text{tr}(\mathbf{\Pi}_j)\epsilon^2} \mathbf{Z} \mathbf{\Pi}_j \mathbf{Z}^\top \right)}_{R^c(\mathbf{Z} | \mathbf{\Pi}, \epsilon)}$$

measures **information gain** for any mixture of subspaces/Gaussians.

The optimal representation **maximizes** the **coding rate reduction** (**MCR**²):

$$\max_{\theta} \Delta R(\mathbf{Z}(\theta), \mathbf{\Pi}, \epsilon) = R(\mathbf{Z}(\theta)) - R^c(\mathbf{Z}(\theta) | \mathbf{\Pi}, \epsilon), \quad \text{s.t. } \mathbf{Z} \subset \mathbb{S}^{d-1}. \quad (3)$$

The whole is **to be maximally**
greater than the sum of the parts!

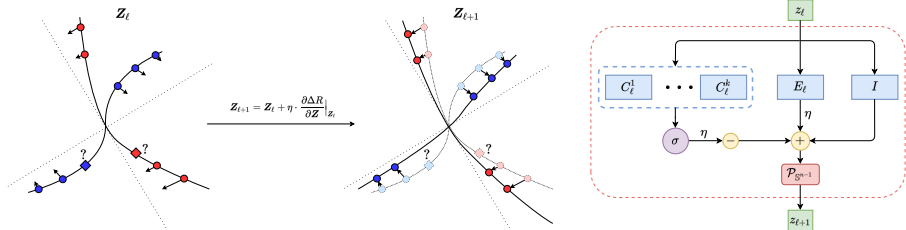


ReduNet: A White-box Deep Network via Rate Reduction

A **white-box**, **forward-constructed**, **multi-channel** (convolution) deep neural network from maximizing the rate reduction via projected gradient flow:

$$\mathbf{Z}_{\ell+1} \propto \mathbf{Z}_{\ell} + \eta \cdot \left. \frac{\partial \Delta R(\mathbf{Z}, \mathbf{\Pi}, \epsilon)}{\partial \mathbf{Z}} \right|_{\mathbf{Z}_{\ell}} \quad \text{s.t.} \quad \mathbf{Z}_{\ell} \subset \mathbb{S}^{d-1}. \quad (4)$$

$$\left. \frac{\partial R(\mathbf{Z})}{\partial \mathbf{Z}} \right|_{\mathbf{Z}_{\ell}} = \underbrace{\alpha(\mathbf{I} + \alpha \mathbf{Z}_{\ell} \mathbf{Z}_{\ell}^*)^{-1} \mathbf{Z}_{\ell}}_{\text{auto-regression residual}} \doteq \mathbf{E}_{\ell} \mathbf{Z}_{\ell} \approx \underbrace{\alpha[\mathbf{Z}_{\ell} - \alpha \mathbf{Z}_{\ell}(\mathbf{Z}_{\ell}^* \mathbf{Z}_{\ell})]}_{\text{self-attention head}}. \quad (5)$$

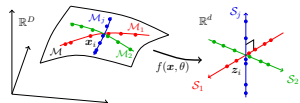


ReduNet: A Whitebox Deep Network from Rate Reduction (JMLR, 2022):

<https://arxiv.org/abs/2105.10446>

White-box Objectives, Architectures, and Representations

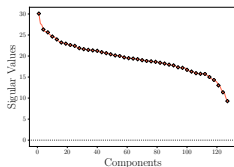
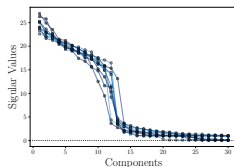
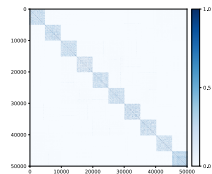
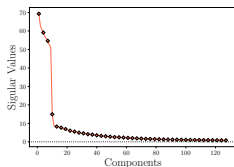
Comparison with conventional practice of NNs (since McCulloch-Pitts'1943).



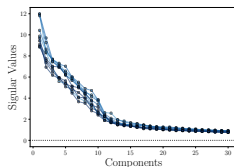
	Conventional DNNs	ReduNets
Objectives	input/output fitting	information gain
Deep architectures	trial & error	iterative optimization
Layer operators	empirical	projected gradient
Shift invariance	CNNs+augmentation	invariant ReduNets
Initializations	random/pre-design	forward unrolled ¹
Training/fine-tuning	back prop	forward/back prop
Interpretability	black box	white box
Representations	hidden/latent	incoherent subspaces

¹ *The Forward-Forward Algorithm*, G. Hinton, 2022.

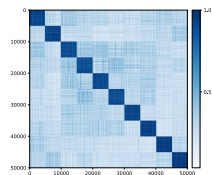
Visualization of Learned Representations \mathcal{Z} (for CIFAR-10)

(a) MCR^2 (overall)(b) MCR^2 (PCA of every class)(c) MCR^2 (cosine similarity)

(d) CE (overall)



(e) CE (PCA of every class)

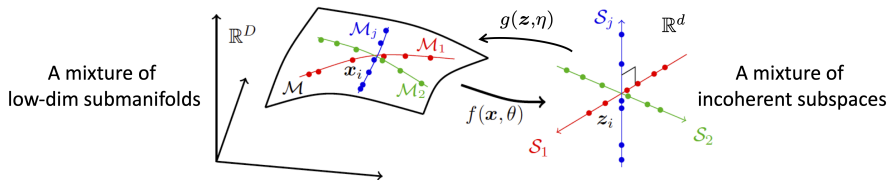


(f) CE (cosine similarity)

Figure: PCA of learned representations from MCR^2 v.s. cross-entropy (CE).

No neural collapse!

Self-Consistency: How to Learn Correctly?



Goal: Transcribe the data $\mathbf{X} \subset \cup_{j=1}^k \mathcal{M}_j$ onto an LDR $\mathbf{Z} \subset \cup_{j=1}^k \mathcal{S}_j$:

$$\underbrace{f(\mathcal{M}_j) = \mathcal{S}_j}_{\text{linear}} \quad \text{with} \quad \underbrace{\mathcal{S}_i \perp \mathcal{S}_j}_{\text{discriminative}} \quad \text{and} \quad \underbrace{g(f(\mathcal{M}_j)) = \mathcal{M}_j}_{\text{auto-embedding}}. \quad (6)$$

Autoencoding of multiple low-dim nonlinear submanifolds:

$$\mathbf{X} \subset \cup_{j=1}^k \mathcal{M}_j \xrightarrow{f(\mathbf{x}, \theta)} \cup_{j=1}^k \mathbf{Z}_j \subset \mathcal{S}_j \xrightarrow{g(\mathbf{z}, \eta)} \hat{\mathbf{X}} \subset \cup_{j=1}^k \mathcal{M}_j. \quad (7)$$

Objective: Compressed & Structured Autoencoding

Desiderata for a **good** representation:

- **Geometry:** f and g are continuous and restricted isometric.
- **Auto-encoding:** f is an embedding of the data \mathbf{X} :

$$g(f(\mathcal{M})) = \mathcal{M}, \quad \text{or} \quad g(f(\mathcal{M}_j)) = \mathcal{M}_j. \quad (8)$$

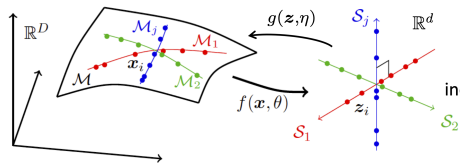
Caveats: we do not know $\dim(\mathcal{M})$ nor $d_j = \dim(\mathcal{M}_j)$. Often

$$d > \dim(\mathcal{M}) \quad \text{or} \quad d > d_1 + d_2 + \cdots + d_k.$$

Structure of the learned $\mathbf{Z} \subset f(\mathcal{M})$ often remains “**hidden**” in \mathbb{R}^d !

- So further wish the feature \mathbf{Z} to be **linear and discriminative**:

$$\begin{aligned} f(\mathcal{M}) &= \mathcal{S} \quad \text{or} \\ f(\mathcal{M}_j) &= \mathcal{S}_j \quad (\text{with } \mathcal{S}_i \perp \mathcal{S}_j). \end{aligned}$$



Self-Consistency: Three Classic Simpler Cases

1. **one low-dim linear subspace:** Principal Component Analysis (**PCA**²)

$$\mathbf{X} \subset \mathcal{S}^D \xrightarrow{\mathbf{V}^T} \mathbf{Z} \subset \mathcal{S}^d \xrightarrow{\mathbf{V}} \hat{\mathbf{X}} \subset \mathcal{S}^D. \quad (9)$$

2. **one low-dim nonlinear submanifold:** **Nonlinear PCA**³

$$\mathbf{X} \subset \mathcal{M}^D \xrightarrow{f(\mathbf{x}, \theta)} \mathbf{Z} \subset \mathcal{S}^d \xrightarrow{g(\mathbf{z}, \eta)} \hat{\mathbf{X}} \subset \mathcal{M}^D. \quad (10)$$

3. **multiple linear subspaces:** Sparsity or Generalized PCA (**GPCA**⁴)

$$\mathbf{X} \subset \cup_{j=1}^k \mathcal{S}_j \xrightarrow{f(\mathbf{x}, \theta)} \cup_{j=1}^k \mathbf{Z}_j \subset \mathcal{S}_j \xrightarrow{g(\mathbf{z}, \eta)} \hat{\mathbf{X}} \subset \cup_{j=1}^k \mathcal{S}_j. \quad (11)$$

The most general, likely the most useful, **real-world case**:

$$\mathbf{X} \subset \cup_{j=1}^k \mathcal{M}_j \xrightarrow{f(\mathbf{x}, \theta)} \cup_{j=1}^k \mathbf{Z}_j \subset \mathcal{S}_j \xrightarrow{g(\mathbf{z}, \eta)} \hat{\mathbf{X}} \subset \cup_{j=1}^k \mathcal{M}_j. \quad (12)$$

²Pearson 1901, Hotelling 1933, Jolliffe 1986.

³Nonlinear PCA using autoassociative neural networks, M. Krammer, 1991.

⁴Generalized principal component analysis, R. Vidal, Yi Ma, and S. Sastry, 2005.

Principal Component Analysis (Auto-Encoding)

One low-dim linear subspace: principal component analysis (PCA)

$$\mathbf{X} \subset \mathcal{S}^D \xrightarrow{\mathbf{V}^T} \mathbf{Z} \subset \mathcal{S}^d \xrightarrow{\mathbf{V}} \hat{\mathbf{X}} \subset \mathcal{S}^D. \quad (13)$$

Solve the following optimization problem:

$$\min_{\mathbf{V}} \|\mathbf{X} - \hat{\mathbf{X}}\|_2^2 \quad \text{s.t.} \quad \hat{\mathbf{X}} = \mathbf{V}\mathbf{V}^T\mathbf{X}, \quad \mathbf{V} \in \mathcal{O}(D, d). \quad (14)$$

Principal Component Analysis (Auto-Encoding)

One low-dim linear subspace: principal component analysis (PCA)

$$\mathbf{X} \subset \mathcal{S}^D \xrightarrow{\mathbf{V}^T} \mathbf{Z} \subset \mathcal{S}^d \xrightarrow{\mathbf{V}} \hat{\mathbf{X}} \subset \mathcal{S}^D. \quad (13)$$

Solve the following optimization problem:

$$\min_{\mathbf{V}} \|\mathbf{X} - \hat{\mathbf{X}}\|_2^2 \quad \text{s.t.} \quad \hat{\mathbf{X}} = \mathbf{V}\mathbf{V}^T\mathbf{X}, \quad \mathbf{V} \in \mathcal{O}(D, d). \quad (14)$$

One low-dim nonlinear submanifold: Nonlinear PCA

$$\mathbf{X} \subset \mathcal{M}^D \xrightarrow{f(\mathbf{x}, \theta)} \mathbf{Z} \subset \mathcal{S}^d \xrightarrow{g(\mathbf{z}, \eta)} \hat{\mathbf{X}} \subset \mathcal{M}^D. \quad (15)$$

Solve the following optimization problem:

$$\min_{\theta, \eta} \underbrace{\|\mathbf{X} - \hat{\mathbf{X}}\|_2^2}_{d(\mathbf{X}, \hat{\mathbf{X}})^2} \quad \text{s.t.} \quad \hat{\mathbf{X}} = g(f(\mathbf{X}, \eta), \theta). \quad (16)$$

What is the right distance $d(\mathbf{X}, \hat{\mathbf{X}})$, say for images?

Auto-Encoding and its Difficulties - Generative Approaches

Nonlinear PCA: Auto-encoding (AE) (Krammer'1991)

$$\mathbf{X} \subset \mathcal{M}^D \xrightarrow{f(\mathbf{x}, \theta)} \mathbf{Z} \subset \mathcal{S}^d \xrightarrow{g(\mathbf{z}, \eta)} \hat{\mathbf{X}} \subset \mathcal{M}^D. \quad (17)$$

Assuming a **generative** model: $p(\mathbf{x}|\mathbf{z}, \Theta)$ and $p(\mathbf{z}, \Theta)$, **maximal likelihood**:


$$\max_{\Theta} P(\mathbf{X}, \Theta) \sim p(\mathbf{x}, \Theta) = \int p(\mathbf{x}|\mathbf{z}, \Theta)p(\mathbf{z}, \Theta)d\mathbf{z}. \quad (18)$$

is in general **intractable**, so is to compute the true posterior

$$P(\mathbf{Z}|\mathbf{X}, \Theta) \sim p(\mathbf{z}|\mathbf{x}, \Theta) = p(\mathbf{x}|\mathbf{z}, \Theta)p(\mathbf{z}, \Theta)/p(\mathbf{x}, \Theta). \quad (19)$$

Instead, optimize certain **variational lower bounds** (VAE):⁵

$$\max -\mathcal{D}_{KL}\left(\underbrace{\hat{p}(\mathbf{z}|\mathbf{x}, \eta)}_{\text{surrogate}}, p(\mathbf{z}, \Theta)\right) + \mathbb{E}_{\hat{p}(\mathbf{z}|\mathbf{x}, \eta)}[\log p(\mathbf{x}|\mathbf{z}, \Theta)]. \quad (20)$$

⁵Auto-Encoding Variational Bayes, D. Kingma and M. Welling, 2014. 

GAN and its Caveats – Discriminative Approaches

Learning generative models via **discriminative** approaches? (Tu'2007)

Generative Adversarial Nets (GAN) (Goodfellow'2014):

$$\mathbf{Z} \xrightarrow{g(\mathbf{z}, \eta)} \hat{\mathbf{X}}, \mathbf{X} \xrightarrow{d(\mathbf{x}, \theta)} \mathbf{0}, \mathbf{1}. \quad (21)$$

A **minimax game** between generator and discriminator:

$$\min_{\eta} \max_{\theta} \mathbb{E}_{p(\mathbf{x})} [\log d(\mathbf{x}, \theta)] + \mathbb{E}_{p(\mathbf{z})} [1 - \log d(\underbrace{g(\mathbf{z}, \eta), \theta}_{\hat{\mathbf{x}} \sim p_g})]. \quad (22)$$

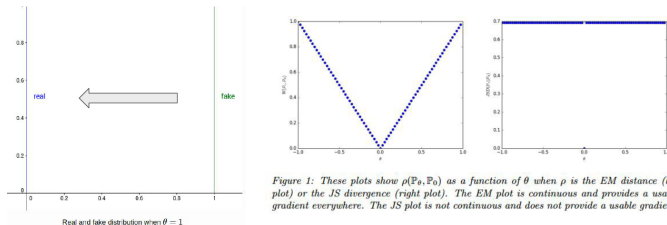
This is equivalent to minimize the *Jensen-Shannon divergence*:

$$\mathcal{D}_{JS}(p, p_g) = \mathcal{D}_{KL}(p \| (p + p_g)/2) + \mathcal{D}_{KL}(p_g \| (p + p_g)/2). \quad (23)$$

**But the J-S divergence is extremely difficult,
if not impossible, to compute and optimize.**

GAN and its Caveats – Discriminative Approaches

J-S distance in high-dim space is like ℓ^0 -norm for distributions with non-overlapping low-dim supports. (**always the case in high-dim!**)



Replace \mathcal{D}_{JS} with the *Earth-Mover* distance or *Wasserstein-1* distance:

$$W_1(p, p_g) = \inf_{\pi \in \Pi(p, p_g)} \mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim \pi} [\|\mathbf{x} - \mathbf{y}\|_1] \quad (24)$$

- **Hard to compute** $\mathcal{D}_{JS}(p, p_g)$ or $W_1(p, p_g)$ accurately and efficiently.
- \mathcal{D}_{JS} or W_1 has **no closed-form** even between two Gaussians!

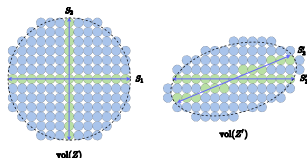
Rate Reduction as Distance between Subspace Gaussians

Rate reduction ΔR gives a **closed-form distance** for (non-overlapping) mixture of subspaces/Gaussians!

$$\Delta R(\mathbf{Z}, \mathbf{\Pi}, \epsilon) = \underbrace{\frac{1}{2} \log \det \left(\mathbf{I} + \frac{d}{m\epsilon^2} \mathbf{Z} \mathbf{Z}^\top \right)}_R - \underbrace{\sum_{j=1}^k \frac{\text{tr}(\mathbf{\Pi}_j)}{2m} \log \det \left(\mathbf{I} + \frac{d}{\text{tr}(\mathbf{\Pi}_j)\epsilon^2} \mathbf{Z} \mathbf{\Pi}_j \mathbf{Z}^\top \right)}_{R^c}.$$

A good measure for the (LDR-like) features \mathbf{Z} , but what about $d(\mathbf{X}, \hat{\mathbf{X}})$?

$$\mathbf{X} \xrightarrow{f(x, \theta)} \mathbf{Z} \xrightarrow{g(z, \eta)} \hat{\mathbf{X}}. \quad (25)$$

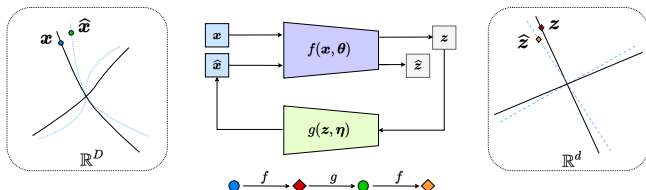


A BIG question:
do we ever need to measure in the external data x space?

Self-Consistency: How to Learn Autonomously?

Is it possible to measure everything **only** in the feature z space?

$$\mathbf{X} \xrightarrow{f(x,\theta)} \mathbf{Z} \xrightarrow{g(z,\eta)} \hat{\mathbf{X}} \xrightarrow{f(x,\theta)} \hat{\mathbf{Z}}. \quad (26)$$



Yes! Measure difference in \mathbf{X}_j and $\hat{\mathbf{X}}_j$ through their features \mathbf{Z}_j and $\hat{\mathbf{Z}}_j$:

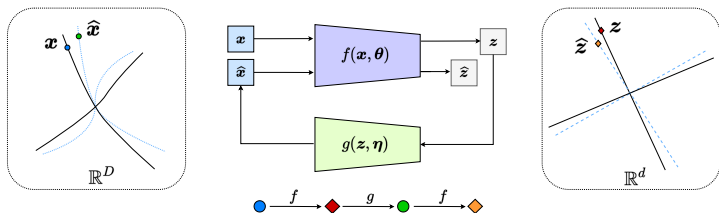
$$\mathbf{X}_j \xrightarrow{f(x,\theta)} \mathbf{Z}_j \xrightarrow{g(z,\eta)} \hat{\mathbf{X}}_j \xrightarrow{f(x,\theta)} \hat{\mathbf{Z}}_j, \quad j = 1, \dots, k. \quad (27)$$

with “their distance” measured by the **rate reduction**:

$$\Delta R(\mathbf{Z}_j, \hat{\mathbf{Z}}_j) \doteq R(\mathbf{Z}_j \cup \hat{\mathbf{Z}}_j) - \frac{1}{2}(R(\mathbf{Z}_j) + R(\hat{\mathbf{Z}}_j)), \quad j = 1, \dots, k. \quad (28)$$

Self-Consistency: Closed-Loop Self-Critiquing Game

Just close the loop and minimize the error is **not enough!**

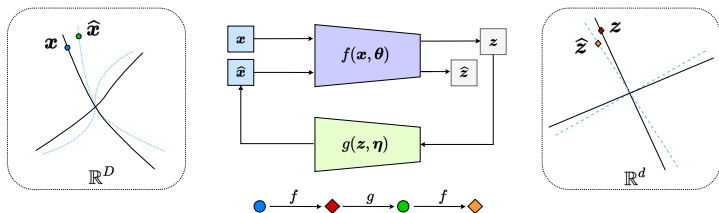


Decoder/generator g minimizes the difference between X and \hat{X} :

$$d(X, \hat{X}) \doteq \min_{\eta} \sum_{j=1}^k \Delta R(Z_j, \hat{Z}_j) = \min_{\eta} \sum_{j=1}^k \Delta R(Z_j, f(g(Z_j, \eta), \theta)).$$

Self-Consistency: Closed-Loop Self-Critiquing Game

Just close the loop and minimize the error is **not enough!**



Decoder/generator g minimizes the difference between \mathbf{X} and $\hat{\mathbf{X}}$:

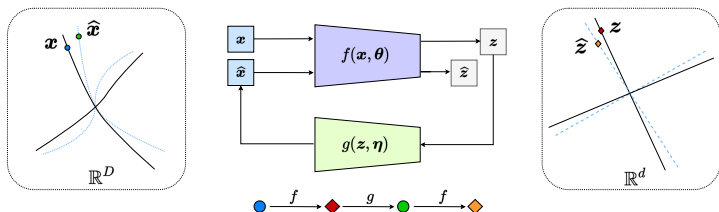
$$d(\mathbf{X}, \hat{\mathbf{X}}) \doteq \min_{\eta} \sum_{j=1}^k \Delta R(\mathbf{Z}_j, \hat{\mathbf{Z}}_j) = \min_{\eta} \sum_{j=1}^k \Delta R(\mathbf{Z}_j, f(g(\mathbf{Z}_j, \eta), \theta)).$$

Encoder/sensor f amplifies any difference between \mathbf{X} and $\hat{\mathbf{X}}$:

$$d(\mathbf{X}, \hat{\mathbf{X}}) \doteq \max_{\theta} \sum_{j=1}^k \Delta R(\mathbf{Z}_j, \hat{\mathbf{Z}}_j) = \max_{\theta} \sum_{j=1}^k \Delta R(f(\mathbf{X}_j, \theta), f(\hat{\mathbf{X}}_j, \theta)).$$

Self-Consistency: Closed-Loop Feedback and Game

f is both an encoder and sensor; and g is both a decoder and controller. They form a closed-loop system for feedback and game:



A closed-loop notion of “**self-consistency**” between Z and \hat{Z} is achieved by a **self-critiquing game** between the sensor f and the generator g :

$$\mathcal{D}(\mathbf{X}, \hat{\mathbf{X}}) \doteq \max_{\theta} \min_{\eta} \sum_{j=1}^k \Delta R \left(\underbrace{f(\mathbf{X}_j, \theta)}_{\mathbf{Z}_j(\theta)}, \underbrace{f(g(f(\mathbf{X}_j, \theta), \eta), \theta)}_{\hat{\mathbf{Z}}_j(\theta, \eta)} \right). \quad (29)$$

Overall Objective: Parsimony & Self-Consistency

The overall **maximin game** between the encoder f and decoder g :

- f *maximizes* the rate reduction of the features \mathbf{Z} of the data \mathbf{X} ;
- g *minimizes* the rate reduction of the features $\hat{\mathbf{Z}}$ of the decoded $\hat{\mathbf{X}}$.

A maximin program to learn a **consistent LDR**⁶ for data $\mathbf{X} = \cup_{j=1}^k \mathbf{X}_j$:

$$\max_{\theta} \min_{\eta} \underbrace{\Delta R(f(\mathbf{X}, \theta))}_{\text{Expansive encode}} + \underbrace{\Delta R(h(\mathbf{X}, \theta, \eta))}_{\text{Compressive decode}} + \sum_{j=1}^k \underbrace{\Delta R(f(\mathbf{X}_j, \theta), h(\mathbf{X}_j, \theta, \eta))}_{\text{Contrastive \& Contractive}}$$

with $h(\mathbf{x}) \doteq f \circ g \circ f(\mathbf{x})$, or equivalently

$$\max_{\theta} \min_{\eta} \Delta R(\mathbf{Z}(\theta)) + \Delta R(\hat{\mathbf{Z}}(\theta, \eta)) + \sum_{j=1}^k \Delta R(\mathbf{Z}_j(\theta), \hat{\mathbf{Z}}_j(\theta, \eta)).$$

⁶CTRL: Closed-Loop Transcription to an LDR via Minimizing Rate Reduction, Entropy, 2022 ([arXiv:2206.09120](https://arxiv.org/abs/2206.09120)).

Characteristics of the Overall Objective

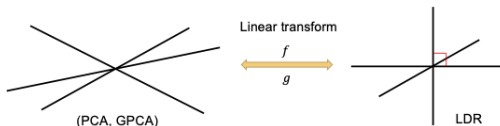
$$\max_{\theta} \min_{\eta} \Delta R(\mathbf{Z}(\theta)) + \Delta R(\hat{\mathbf{Z}}(\theta, \eta)) + \sum_{j=1}^k \Delta R(\mathbf{Z}_j(\theta), \hat{\mathbf{Z}}_j(\theta, \eta)).$$

- **Parsimony:** all terms are **closed-form** rate reduction on features.
- **Self-consistency:** enforced by **closed-loop** feedback and game.
- **Structured:** distribution of learned features \mathbf{Z} is **an LDR**.
- **No** need to specify a prior or a surrogate target distribution.
- **No** need of any direct explicit distance between \mathbf{X} and $\hat{\mathbf{X}}$.
- **No** more approximations or bounds for (KL-, JS-, W-) “distances”.
- **No** heuristics or regularizing terms in the objective.

Parsimony and self-consistency are all you need to model \mathbf{X} ?⁷

⁷CTRL: Closed-Loop Transcription to an LDR via Minimizing Rate Reduction, Entropy, 2022 ([arXiv:2206.09120](https://arxiv.org/abs/2206.09120)).

Theoretical Guarantee for the Case of Multiple Subspaces



Theorem (Stackelberg Equilibria of CTRL Multi-Subspace Pursuit (arXiv:2206.09120))

The CTRL-MSP game has a Stackelberg equilibrium, and all Stackelberg equilibria (f_\star, g_\star) have the following properties:

- ① (Injective encoder) For each $j \in \{1, \dots, k\}$, we have that $f_\star(S_j)$ is a linear subspace of dimension d_{S_j} , and one of the following holds:
 - $\sigma_1(f_\star(\mathbf{X}_j)) = \dots = \sigma_{d_{S_j}}(f_\star(\mathbf{X}_j)) = \frac{n_j}{d_{S_j}}$; or
 - $\sigma_1(f_\star(\mathbf{X}_j)) = \dots = \sigma_{d_{S_j}-1}(f_\star(\mathbf{X}_j)) \in (\frac{n_j}{d_{S_j}}, \frac{n_j}{d_{S_j}-1})$ and $\sigma_{d_{S_j}}(f_\star(\mathbf{X}_j)) > 0$, where if $d_{S_j} = 1$ then $\frac{n_j}{d_{S_j}-1}$ is interpreted as $+\infty$.
- ② (Discriminative encoder) The subspaces $\{f_\star(S_j)\}_{j=1}^k$ are orthogonal.
- ③ (Consistent encoding and decoding) For each $j \in \{1, \dots, k\}$, we have that $f_\star(S_j) = (f_\star \circ g_\star \circ f_\star)(S_j)$.

Empirical Verification of CTRL on Visual Data

Experimental Setup:

- **Datasets:** MNIST, CIFAR10, STL-10, CelebA faces, LSUN bedroom, ImageNet
- **Network architectures:** basic DCGAN & ResNet (**not customized**).
- **Feature space:** **the same** 128-dim regardless of data resolution or size
- **Quantization precision:** **the same** $\epsilon^2 = 0.5$.
- **Optimizer:** *Adam* with **the same** hyperparameters $\beta_1 = 0, \beta_2 = 0.9$.
- **Linear rate:** **the same** initial 0.00015 with linear decay.

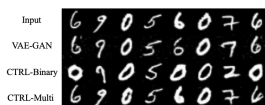
No other regularization, heuristics, or engineering tricks.⁸

⁸*CTRL: Closed-Loop Transcription to an LDR via Minimizing Rate Reduction*, Entropy, 2022 ([arXiv:2206.09120](https://arxiv.org/abs/2206.09120)).

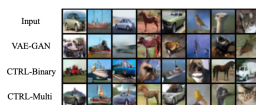
Empirical Verification: Fair Comparison to Baselines

Method		GAN	GAN (CTRL)	VAE-GAN	CTRL-Binary	CTRL-Multi
MNIST	IS \uparrow	2.08	1.95	2.21	2.02	2.07
	FID \downarrow	24.78	20.15	33.65	16.43	16.47
CIFAR-10	IS \uparrow	7.32	7.23	7.11	8.11 (8.4)	7.13 (8.2)
	FID \downarrow	26.06	22.16	43.25	19.63 (18.7)	23.91 (20.5)

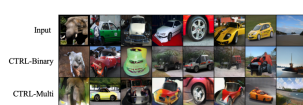
Table: Quantitative comparison on MNIST and CIFAR-10. Average Inception scores (IS) and FID scores. \uparrow means higher is better. \downarrow means lower is better.



(a) MNIST



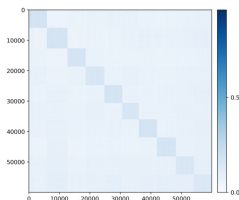
(b) CIFAR-10



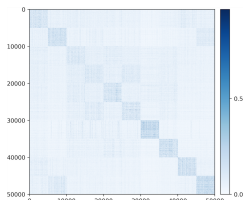
(c) ImageNet

Figure: Qualitative comparison on MNIST, CIFAR-10 and ImageNet.

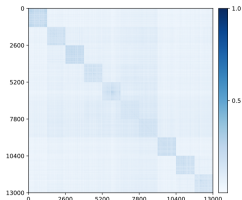
Empirical Verification on Visual Data



(a) MNIST



(b) CIFAR10



(c) ImageNet

Figure: Visualizing the alignment between Z and \hat{Z} : $|Z^\top \hat{Z}|$.

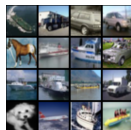
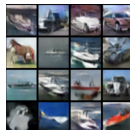
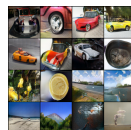
(a) MNIST X (b) MNIST \hat{X} (c) CIFAR10 X (d) CIFAR10 \hat{X} (e) ImageNet X (f) ImageNet \hat{X}

Figure: Visualizing the auto-encoding property: $x \approx \hat{x} = g \circ f(x)$.

Empirical Verification: MNIST PCAs

The feature z in each of the k principal subspaces can be modeled as a degenerate Gaussian from the PCA $Z_j = V_j \Sigma_j U_j^T$:

$$z_j \sim \bar{z}_j + \sum_{l=1}^{r_j} n_l^j \sigma_j^l v_j^l, \quad \text{where } n_l^j \sim \mathcal{N}(0, 1), \quad j = 1, \dots, k. \quad (30)$$



(a) ACGAN



(b) InfoGAN



(c) CTRL-Multi

Empirical Verification: MNIST PCAs

The feature z in each of the k principal subspaces can be modeled as a degenerate Gaussian from the PCA $Z_j = V_j \Sigma_j U_j^T$:

$$z_j \sim \bar{z}_j + \sum_{l=1}^{r_j} n_l^j \sigma_j^l v_j^l, \quad \text{where} \quad n_l^j \sim \mathcal{N}(0, 1), \quad j = 1, \dots, k. \quad (31)$$

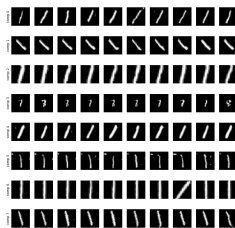
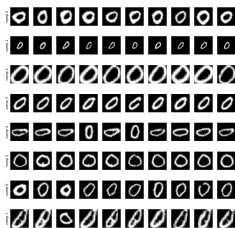
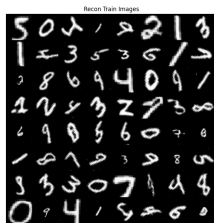
Nearest subspace classification based on the computed PCAs.

Method	VAE	Factor VAE	Guide-VAE	DC-VAE	CTRL-Binary	CTRL-Multi
MNIST	97.12%	93.65%	98.51%	98.71%	89.12%	98.30%

Table: Classification accuracy on MNIST, comparing to classifier based VAE methods. Most of those VAE-based methods require auxiliary classifiers to boost classification performance.

Empirical Verification: Transformed MNIST

Original data \mathbf{X} and their decoded version $\hat{\mathbf{X}}$ on transformed MNIST.

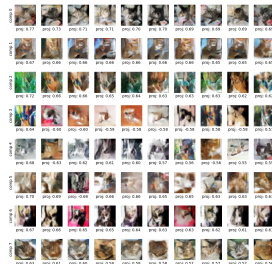
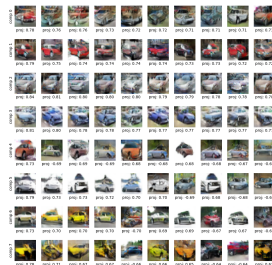
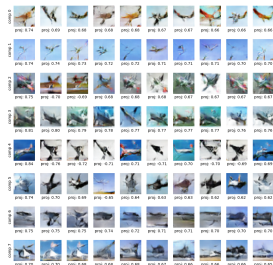


(c) Components of "0"

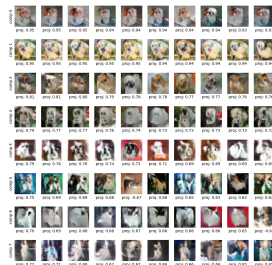
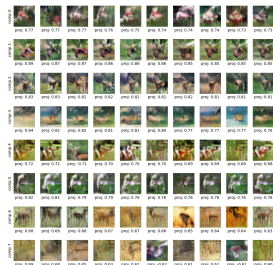
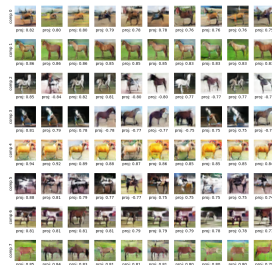
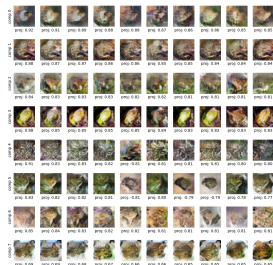
(d) Components of "1"

(e) Components of "2"

Empirical Verification: “Principal Images” of CIFAR10



Empirical Verification: “Principal Images” of CIFAR10



Empirical Verification: “Principal Images” of CIFAR10

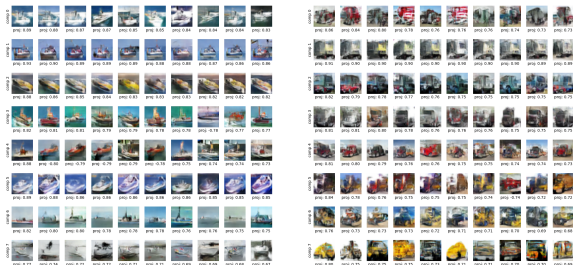
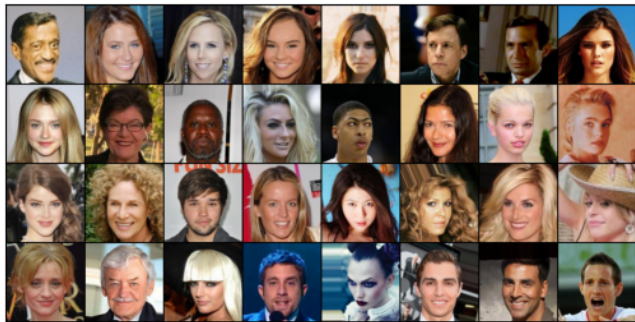


Figure: Reconstructed images \hat{X} from features Z close to the principal components learned for each of the 10 classes of CIFAR-10.

Different classes are disentangled as principal subspaces.
Visual attributes are disentangled as principal components.

Empirical Verification: CelebA Input X



(a) Original X

Figure: Visualizing the original x and corresponding decoded \hat{x} results on Celeb-A dataset. The LDR model is trained from CTRL-Binary.

Empirical Verification: CelebA Decoded \hat{X}

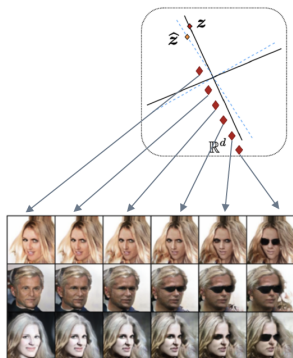


(a) Decoded \hat{X}

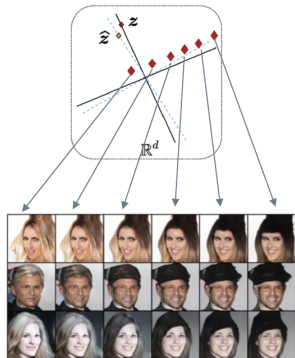
Figure: Visualizing the original x and corresponding decoded \hat{x} results on Celeb-A dataset. The LDR model is trained from CTRL-Binary.

Empirical Verification: Principal Components of CelebA

Figure: Generated images by sampling along the 9-th and 23-th principal components of the learned features \mathbf{Z} , for the CelebA dataset.



Generated along one PC



Generated along another PC

Visual attributes are disentangled as principal components.

Empirical Verification: Ablation Study

Training the ImageNet with networks of different width.

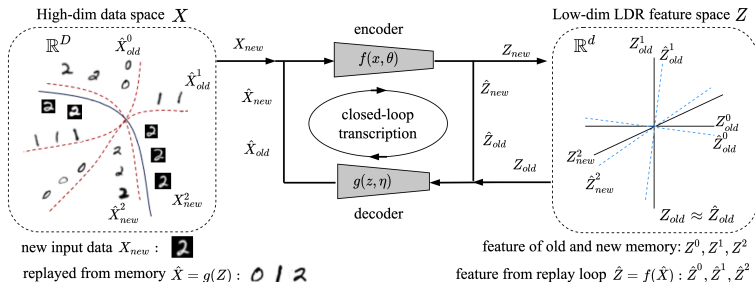
	channel#=1024	channel#=512	channel#=256
BS=1800	success	success	success
BS=1600	success	success	success
BS=1024	failure	success	success
BS=800	failure	failure	success
BS=400	failure	failure	failure

Table: Ablation study on ImageNet about tradeoff between batch size (BS) and network width (channel #).

No mode collapse!

Incremental Learning via Closed-Loop Transcription

Incremental Learning of Structured Memory: one class at a time.⁹



$$\begin{aligned}
 & \max_{\theta} \min_{\eta} \quad \Delta R(\mathbf{Z}) + \Delta R(\hat{\mathbf{Z}}) + \Delta R(\mathbf{Z}_{new}, \hat{\mathbf{Z}}_{new}) \\
 & \text{subject to} \quad \Delta R(\mathbf{Z}_{old}, \hat{\mathbf{Z}}_{old}) = 0.
 \end{aligned} \tag{32}$$

⁹Incremental Learning of Structured Memory via Closed-Loop Transcription, S. Tong and Yi Ma et. al., ICLR 2023. ([arXiv:2202.05411](https://arxiv.org/abs/2202.05411))

Incremental Learning via Closed-Loop Transcription

Incremental Learning of Structured Memory: one class at a time.¹⁰

Method	MNIST	CIFAR10
INFORS (Sun et. al., ICLR 2022)	0.814	0.526
CLS-ER (Arani et. al. ICLR 2022)	0.895	0.662
i-LDR(ours)	0.990	0.723

Table: Comparison with latest SOTA on MNIST and CIFAR-10.

iCaRL-S	EEIL-S	DGMw	EEC	EECS	i-LDR
0.290	0.118	0.178	0.352	0.309	0.523

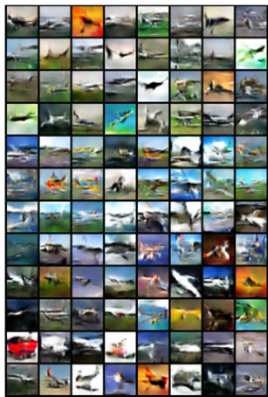
Table: Comparison on ImageNet-50. The results of other methods are as reported in the EEC paper.

No catastrophic forgetting!

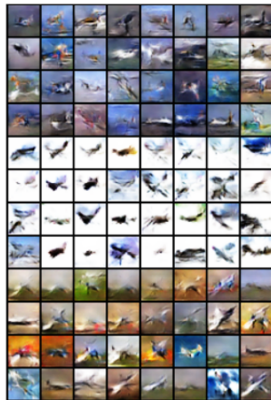
¹⁰Incremental Learning of Structured Memory via Closed-Loop Transcription, S. Tong and Yi Ma et. al., ICLR 2023. ([arXiv:2202.05411](https://arxiv.org/abs/2202.05411))

Incremental Learning via Closed-Loop Transcription

Memory consolidation via review (ICLR 2023)



(a) \hat{x}_{old} before review

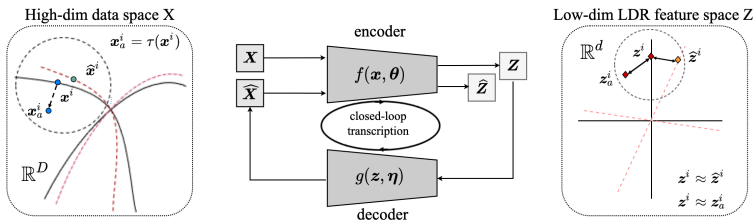


(b) \hat{x}_{old} after review

Figure: Visualization of replayed images \hat{x}_{old} of class 1-‘airplane’ in CIFAR10.

Unsupervised Learning via Closed-Loop Transcription

Unsupervised Learning of Structured Memory: one sample at a time¹¹



$$\max_{\theta} \min_{\eta} R(\mathbf{Z}) + \Delta R(\mathbf{Z}, \hat{\mathbf{Z}}) \quad (33)$$

$$\text{subject to } \sum_{i \in N} \Delta R(z^i, \hat{z}^i) = 0, \text{ and } \sum_{i \in N} \Delta R(z^i, z_a^i) = 0.$$

¹¹Unsupervised Learning of Structured Representations via Closed-Loop Transcription, S. Tong, Yann LeCun, and Yi Ma, [arXiv:2210.16782](https://arxiv.org/abs/2210.16782), 2022.

Unsupervised Learning via Transcription (on CIFAR-10)

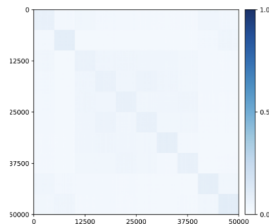
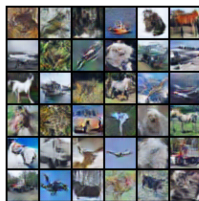
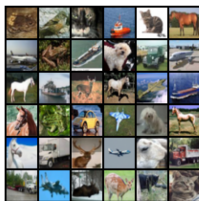


Figure: Sample-wise self-consistency and block-diagonal structures.

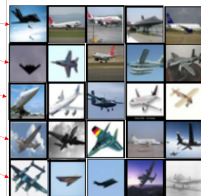
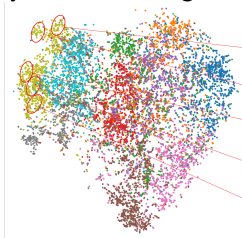
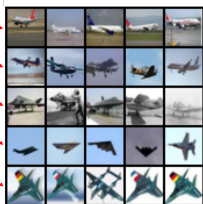
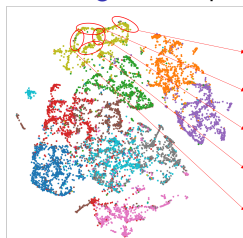
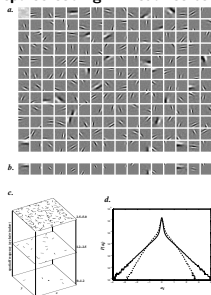


Figure: t-SNE of learned features . Left: U-CTRL and Right: MoCoV2.

Structured Memory in Nature

- Sparse coding in visual cortex (Olshausen, Nature 1996)¹².
- Subspace embedding (Tsao, Cell 2017, Nature 2020).¹³
- Predictive coding in visual cortex (Rao, Nature Neuroscience 1999).

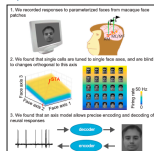
sparse coding in visual cortex



Cell

The Code for Facial Identity in the Primate Brain

Graphical Abstract



Highlights

- Facial images can be linearly reconstructed using responses of ~200 face cells
- Face cells display flat tuning along dimensions orthogonal to the axis being coded
- The axis model is more efficient, robust, and flexible than the exemplar model
- Face patches MLNF and AM carry complementary information about faces

Article

Authors

Le Chang, Doris Y. Tsao

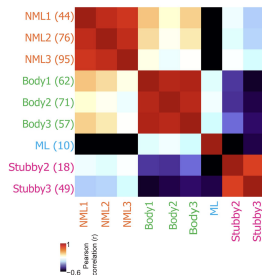
Correspondence
lechang@caltech.edu (L.C.),
dortso@caltech.edu (D.Y.T.)

In Brief

Facial identity is encoded via a remarkably simple neural code that relies on the ability of neurons to distinguish facial features along specific axes in face space, discovering the long-standing assumption that single face cells encode individual faces.

d

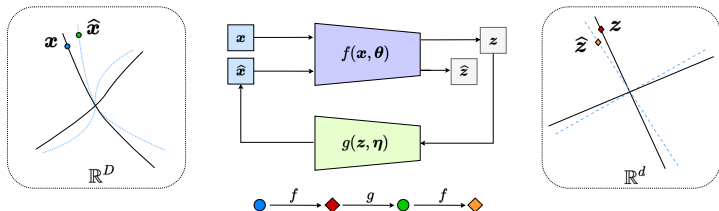
Similarity matrix of the response profile from each area



¹²Figure from Bruno Olshausen of Neuroscience Dept., UC Berkeley.

¹³Figures from Doris Tsao of Neuroscience Dept., UC Berkeley

Conclusions: Compressive Closed-Loop Transcription



- **a universal learning engine:** transform sensed data of external world to a compact and structured (LDR or sparse) internal representation.
- **parsimony:** optimization of the information gain (rate reduction) via a sensor and a generator.
- **self-consistency:** a self-critiquing game between the sensor and generator through a closed-loop feedback system.
- **a white-box system:** learning objectives, network architectures & operators, and learned representations.

Open Mathematical Problems

For the closed-loop maximin rate reduction program:

$$\max_{\theta} \min_{\eta} \Delta R(\mathbf{Z}(\theta)) + \Delta R(\hat{\mathbf{Z}}(\theta, \eta)) + \sum_{j=1}^k \Delta R(\mathbf{Z}_j(\theta), \hat{\mathbf{Z}}_j(\theta, \eta)).$$

- **optimality:** characterization of the **equilibrium points**?
- **convergence** of the closed-loop control problem (**infinite-dim**)?
- **linearization** of distribution supports (**plastic manifold learning**)?
- **optimal density** of the distributions (***Brascamp-Lieb inequalities***)?
- **correct model selection** (**no under- or over-fitting**)?
- **guarantees** for approximate **distribution/sample-wise auto-encoding**?

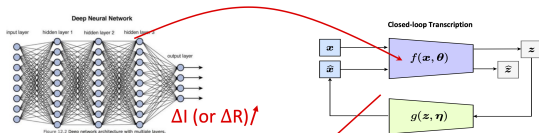
Open Directions: Extensions and Connections

- how to **scale up** to hundreds and thousands of classes?
(variational forms for rate reduction...)
- **whitebox** architectures for closed-loop transcription (ReduNet like)?
- **learning dictionaries** to learn patterns in 1D sequence, 2D image, or 3D space (transformers...)?
- computational mechanisms for **memory** forming (in Nature)?
(recognition and generation integrated...)
- closed-loop transcription for **other types of low-dim structures**?
(dynamical, causal, logical, symbolical, graphical, genetic...)

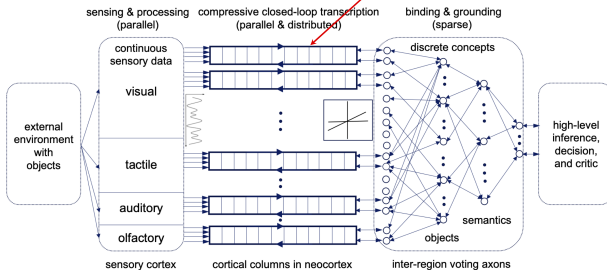
The principles of **parsimony and self-consistency** shall always rule!

World Model: An Integrated System of Transcriptions?

Neural networks are nature's **optimization** algorithms that maximize information gain. (one iteration per layer)



Closed-loop transcriptors are basic learning units for **autonomous** self-consistency. (error feedback & self-critique)



Robustly and efficiently learn compact structured representations of the world. (parallel & distributed)



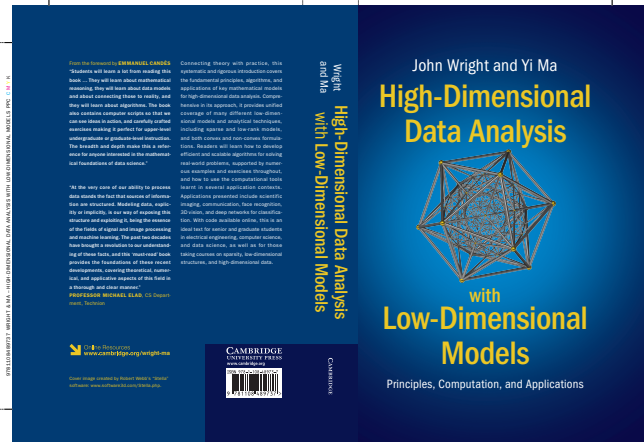
A unified purpose: maximize "information gain" with every unit, at every stage!

References: Closed-Loop Transcription via Rate Reduction

- ① **Principles of Parsimony and Self-Consistency** for Emergence of Intelligence
<https://arxiv.org/abs/2207.04630> (FITEE 2022)
- ② **CTRL: Closed-Loop Transcription to an LDR via Minimizing Rate Reduction**
<https://arxiv.org/abs/2111.06636> (Entropy 2022)
- ③ **Unsupervised Learning** of Structured Memory via Closed-Loop Transcription
<https://arxiv.org/abs/2210.16782>
- ④ **Incremental Learning** of Structured Memory via Closed-Loop Transcription
<https://arxiv.org/abs/2202.05411> (ICLR 2023)

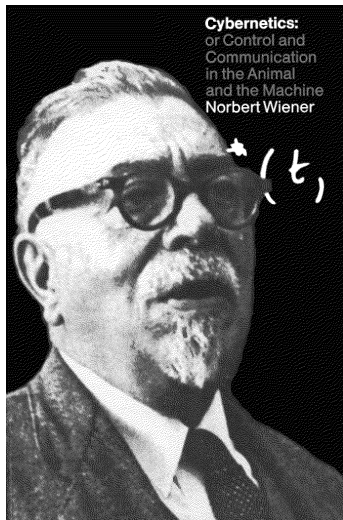
Textbook: *High-Dim Data Analysis*, Wright & Ma, 2022<https://book-wright-ma.github.io/>

Sparse coding
 Low-dim models
 Error correction
 Optimization
 Compression
 Nonlinearity
 Deep networks
 ...



The Classic: *Cybernetics*, Norbert Wiener, 1948/1961

Compact coding
 Closed-loop feedback
 Learning via games
 White-box modeling
 Nonlinearity
 Shift-invariance
 Brain waves
 ...



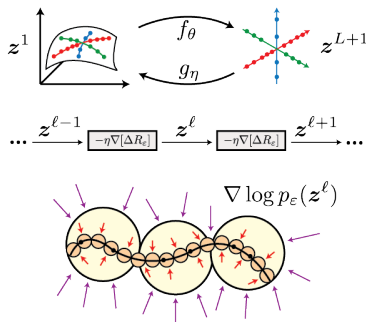
A New Open-Source Textbook on GitHub

Learning Deep Representations of Data Distributions

Sam Buchanan, Druv Pai, Peng Wang, and Yi Ma

Version 1.0, August 18, 2025.

<https://ma-lab-berkeley.github.io/deep-representation-learning-book/>



**Closed-loop transcription is a universal learning machine
for autonomously learning consistent representations.**

Thank you!
Questions, please?

*“What I cannot create, I do not understand.”
– Richard Feynman (his last words)*



SIMONS
FOUNDATION