

Deep Representations via Unrolled Optimization

Druv Pai

UC Berkeley



Lectures so Far

- **History** of the pursuit and study of intelligence.
- **Analytic solutions** for learning low-dimensional linear/Gaussian mixtures via (unrolled) optimization.
- **Learning and sampling** via denoising and compression.
- **Objectives for representation learning**: information gain.

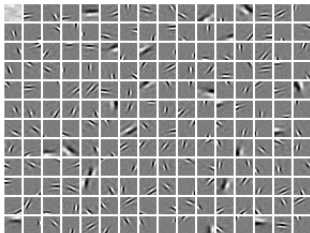
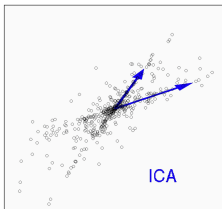
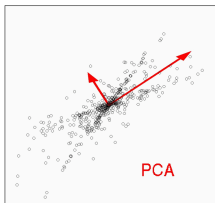
This lecture: deep representation learning!

Operationalize these principles to inform deep network architectures (with empirical success!)

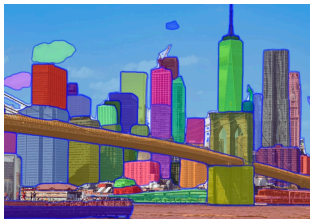
Recall: Classical Approaches

Assume: data drawn from a well-known template distribution:

- A single subspace (PCA)
- A union of a few subspaces (ICA)
- Sparsely generated from a dictionary (DL)



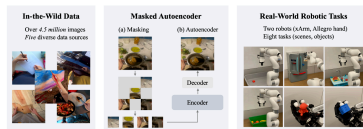
From Analytical to General



Understanding and interacting with the physical world

⇒ **nonlinear signals!**

Nonlinearity demands more flexible representations.



Recall: LISTA

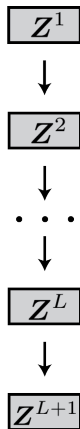
Solve the **sparse representation** problem

$$\min_{\mathbf{Z} \geq \mathbf{0}} [\|\mathbf{AZ} - \mathbf{X}\|_F^2 + \lambda \|\mathbf{Z}\|_1]$$

via **unrolled optimization** (LISTA [GL10]):

$$\begin{aligned}\mathbf{Z}^{\ell+1} &= \text{prox}_{\lambda \|\cdot\|_1 + \chi_{\mathbb{R}_+^d}}(\mathbf{Z}^\ell - \kappa \nabla_{\mathbf{Z}^\ell} \|\mathbf{A}^\ell \mathbf{Z}^\ell - \mathbf{X}\|_F^2) \\ &= \text{ReLU}(\mathbf{Z}^\ell - \kappa \mathbf{A}^{\ell\top} (\mathbf{A}^\ell \mathbf{Z}^\ell - \mathbf{X}) - \kappa \lambda \mathbf{1}) \\ &= \text{ISTA}_{\kappa, \lambda}(\mathbf{Z}^\ell \mid \mathbf{A}^\ell)\end{aligned}$$

Learn the parameters $(\mathbf{A}^\ell)_{\ell=1}^L$ using data.



Each update **incrementally optimizes** the features!

LISTA is a prototypical deep network!

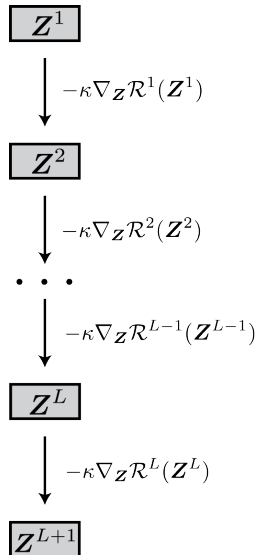
Unrolled Optimization

- Given **objective function** \mathcal{R}^ℓ , improve it on input \mathbf{Z}^ℓ via **optimization step**:

$$\mathbf{Z}^{\ell+1} \leftarrow \mathbf{Z}^\ell - \kappa \nabla_{\mathbf{Z}} \mathcal{R}^\ell(\mathbf{Z}^\ell)$$

(...or similar).

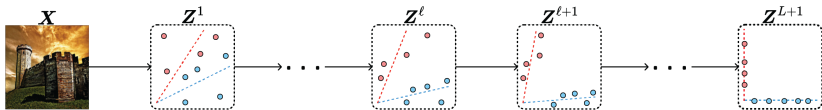
- Collection of objective functions $(\mathcal{R}^\ell)_{\ell=1}^L$ + optimization strategies \Rightarrow **data processing algorithm**
- New: collection of objective functions + optimization strategies \Rightarrow **deep network architecture!**



From Unrolled Optimization to Deep Architectures

Constructing deep network architectures:

Design objectives \mathcal{R}^ℓ and optimization strategies s.t.
unrolling yields **compact, structured, deep representation!**



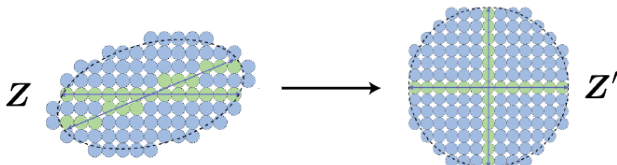
This talk: Use this principle to explain **existing** architectures.

Next talk: Use this principle to build **novel** architectures.

Recall: Rate Reduction

Difference in *coding rate* between encoding data as **whole** (Gaussian) and as **parts** (class-wise GMM):

$$\Delta R(\mathbf{Z} \mid \Pi) = \underbrace{\frac{1}{2} \log \det(\mathbf{I} + \alpha \mathbf{Z} \mathbf{Z}^\top)}_{R(\mathbf{Z}) = \text{blue}} - \underbrace{\frac{1}{2} \sum_{k=1}^K \gamma_k \log \det(\mathbf{I} + \alpha_k \mathbf{Z} \Pi_k \mathbf{Z}^\top)}_{= R^c(\mathbf{Z} \mid \Pi) = \text{green}}$$



$$\Delta R(\mathbf{Z} \mid \Pi) < \Delta R(\mathbf{Z}' \mid \Pi)$$

ΔR = the **information gain** of the representation.

Goal: **Maximize information gain!**

Formulation: $\max_{\theta} \Delta R(\mathbf{Z}(\theta) \mid \Pi)$, where $\mathbf{Z}(\theta) = f(\mathbf{X}, \theta)$.

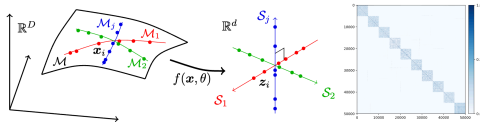
Recall: Global Optimizers of Rate Reduction

Theorem ([Yu+20], Theorem 2.1). Consider the problem

$$\max_{\mathbf{Z}} \Delta R(\mathbf{Z} \mid \Pi) \quad \text{s.t.} \quad \forall k: \|\mathbf{Z}_k\|_F^2 \leq m_k, \text{rank}(\mathbf{Z}_k) \leq d_k.$$

Any (global) maximizer $\mathbf{Z}^* = \bigcup_{k=1}^K \mathbf{Z}_k^*$ satisfies:

- **Between-class discriminative:** If $d \geq \sum_{k=1}^K d_k$, then \mathbf{Z}_k^* are pairwise orthogonal: $\mathbf{Z}_k^{*\top} \mathbf{Z}_j^* = \mathbf{0}$ for all j, k .
- **Maximally diverse representation:** If $\varepsilon < \min_k \left\{ \frac{m_k}{m} \frac{d^2}{d_k^2} \right\}$, each subspace takes its maximal dimension, $\text{rank}(\mathbf{Z}_k^*) = d_k$, and the largest $d_k - 1$ singular values of \mathbf{Z}_k^* are all equal.



Local Optimizers of Rate Reduction

Theorem ([Wan+24], Theorem 1, Simplified.) Consider the problem

$$\max_{\mathbf{Z}} \left\{ \Delta R(\mathbf{Z} \mid \mathbf{\Pi}) - \frac{\lambda}{2} \|\mathbf{Z}\|_F^2 \right\}.$$

If $\lambda \in \left(0, \frac{d(\sqrt{m/m_{\max}}-1)}{m\varepsilon^2(\sqrt{m/m_{\max}}+1)} \right)$, then:

1. $\mathbf{Z}^* = \bigcup_{k=1}^K \mathbf{Z}_k^*$ is a *local* maximizer if and only if: (1) $\text{rank}(\mathbf{Z}_k^*) < \min(d, m_k)$ and $\sum_{k=1}^K \text{rank}(\mathbf{Z}_k^*) \leq d$; (2) the \mathbf{Z}_k^* are **pairwise orthogonal**; and (3) **all singular values of \mathbf{Z}_k^* are equal**.
2. \mathbf{Z}^* is a global maximizer if and only if (1) all above conditions hold, (2) $\sum_{k=1}^K \text{rank}(\mathbf{Z}_k^*) = \min\{m, d\}$, (3) all but the largest class has $\text{rank}(\mathbf{Z}_k^*) = \min\{m_k, d\}$.

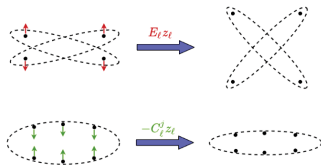
Projected Gradient Ascent on Information Gain

Simplest possible optimization strategy (PGA)

$$\mathbf{Z}^{\ell+1} = \mathcal{P}_{(\mathbb{S}^{d-1})^m} \left(\mathbf{Z}^{\ell} + \kappa \nabla_{\mathbf{Z}} [\Delta R(\mathbf{Z}^{\ell} \mid \mathbf{\Pi})] \right)$$

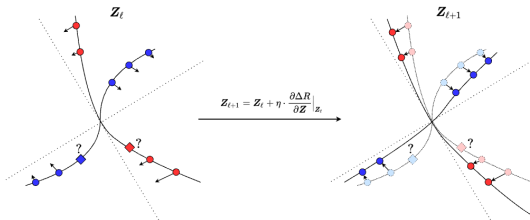
$$\nabla_{\mathbf{Z}} R(\mathbf{Z}^{\ell}) = \underbrace{\alpha (\mathbf{I} + \alpha \mathbf{Z}^{\ell} \mathbf{Z}^{\ell \top})^{-1}}_{\doteq \mathbf{E}^{\ell}} \mathbf{Z}^{\ell}$$

$$\nabla_{\mathbf{Z}} R^c(\mathbf{Z}^{\ell} \mid \mathbf{\Pi}) = \sum_{k=1}^K \gamma_k \underbrace{\alpha_k (\mathbf{I} + \alpha_k \mathbf{Z}^{\ell} \mathbf{\Pi}_k \mathbf{Z}^{\ell \top})^{-1}}_{\doteq \mathbf{C}_k^{\ell}} \mathbf{Z}^{\ell} \mathbf{\Pi}_k$$



$$\nabla_{\mathbf{Z}} [\Delta R(\mathbf{Z}^{\ell} \mid \mathbf{\Pi})] =$$

$$\mathbf{E}^{\ell} \mathbf{Z}^{\ell} - \sum_{k=1}^K \gamma_k \mathbf{C}_k^{\ell} \mathbf{Z}^{\ell} \mathbf{\Pi}_k$$



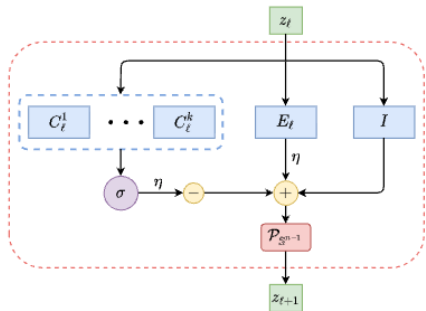
ReduNet

At inference time, estimate $\mathbf{\Pi}$ via softmax on residuals:

$$\hat{\pi}_{i,k} := \frac{\exp(-\lambda \|\mathbf{C}_k^\ell \mathbf{z}_i^\ell\|_2)}{\sum_{j=1}^K \exp(-\lambda \|\mathbf{C}_j^\ell \mathbf{z}_i^\ell\|_2)}, \quad \hat{\mathbf{\Pi}}_k = \text{diag}([\hat{\pi}_{1,k}, \dots, \hat{\pi}_{m,k}]).$$

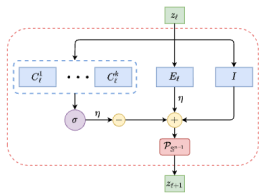
ReduNet arch.: parameters $(\mathbf{E}^\ell, (\mathbf{C}_k^\ell)_{k=1}^K)_{\ell=1}^L$ and forward pass

$$\mathbf{Z}^{\ell+1} = \mathcal{P}_{(\mathbb{S}^{d-1})^m}(\mathbf{Z}^\ell + \kappa \{ \mathbf{E}^\ell \mathbf{Z}^\ell - \underbrace{\sigma([\mathbf{C}_1^\ell \mathbf{Z}^\ell, \dots, \mathbf{C}_K^\ell \mathbf{Z}^\ell])}_{:= \sum_{k=1}^K \gamma_k \mathbf{C}_k^\ell \mathbf{Z}^\ell \hat{\mathbf{\Pi}}_k} \})$$

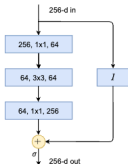


This is a **white-box deep neural network** designed to incrementally optimize the **information gain** of the representation.

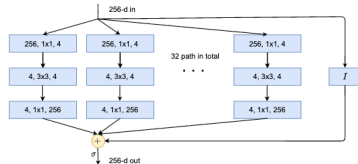
ReduNet and CNNs



[Cha+22]



[He+16]



[Xie+17]

- Pack all *translations* of samples in \mathbf{Z}^ℓ into a block-circulant matrix $\tilde{\mathbf{Z}}^\ell = [\text{circ}(\mathbf{z}_1^\ell), \dots, \text{circ}(\mathbf{z}_m^\ell)] \in \mathbb{R}^{d \times dm}$.
- ReduNet construction \implies matrices $\tilde{\mathbf{E}}^\ell = \text{circ}(\mathbf{k}^\ell)$ and $\tilde{\mathbf{C}}_k^\ell$ are circulant.
- Circulant multiplication $\tilde{\mathbf{E}}^\ell \tilde{\mathbf{Z}}^\ell = \text{convolution } \mathbf{k}^\ell * \mathbf{Z}^\ell! \implies \text{CNN!}$

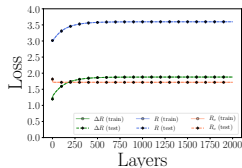
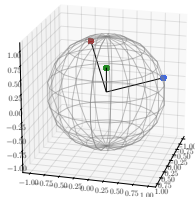
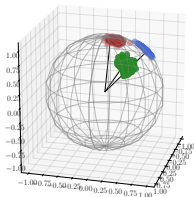
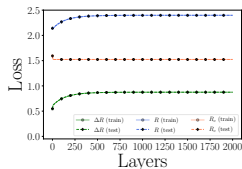
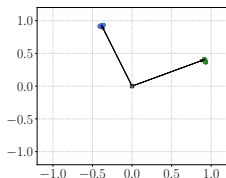
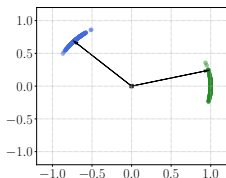
Circulant matrix:

$\text{circ}(\mathbf{k}) =$

$$\begin{bmatrix} k(1) & k(2) & \cdots & k(d) \\ k(2) & k(3) & \cdots & k(1) \\ \vdots & \vdots & \ddots & \vdots \\ k(d) & k(1) & \cdots & k(d-1) \end{bmatrix}$$

Visualizing ReduNet

$L = 2000$ -Layers ReduNet: $m = 500, \kappa = 0.5, \epsilon = 0.1$.



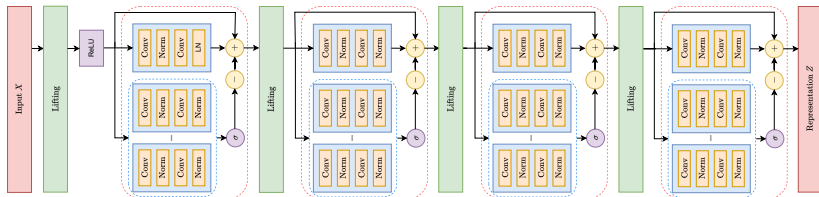
Gaussian mixtures in 2D (top) and 3D (bottom) processed by ReduNet.

ReduNet Summary

ReduNet yields a **white-box, forward-constructable, multi-channel (convolutional) deep neural network** via

explicitly pursuing low-dimensional structures in x and z !

$$f: X \xrightarrow{f^{\text{pre}}} Z^1 \rightarrow \dots \rightarrow Z^\ell \xrightarrow{f^\ell} Z^{\ell+1} \rightarrow \dots \xrightarrow{f^L} Z^{L+1} = Z$$



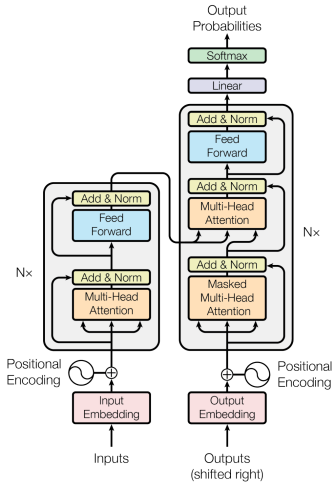
$$Z^{\ell+1} = f^\ell(Z^\ell) \approx Z^\ell + \kappa \nabla [\Delta R(Z^\ell | \Pi)]$$

Modernizing ReduNet

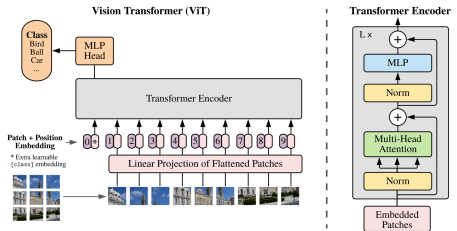
- Initialization: forward-construction needs matrix inverse (not scalable); random init. \sim naive CNN/ResNet.
- Optimization: Requires batch statistics \implies need large batch size to have stable training.
- Representation: one low-dimensional vector per sample \implies poor on dense tasks (segmentation, detection).

Our transformer-like architecture **CRATE** fixes these issues!

Transformers: Modern Deep Learning's Workhorse

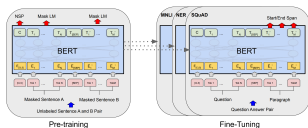


[Vas+17]

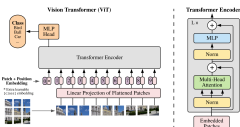


[Dos+20]

Transformers: A Universal Backbone



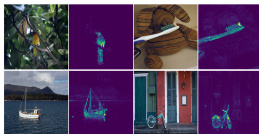
BERT [Dev+19]



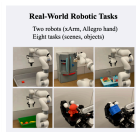
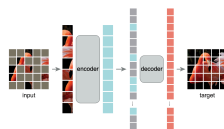
ViT [Dos+20]



GPT [Rad+19]



DINO [Car+21]



Robot Learning w/ MAE

[Rad+23]

TF + NLP

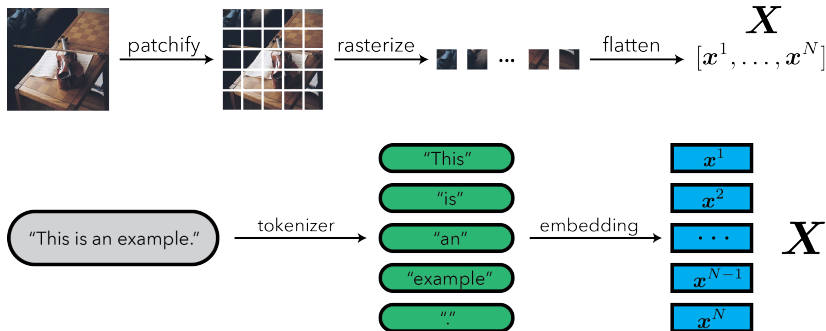
TF + Vision

TF + Robotics

Scaling Data Processing

A more modern data format: sequences!

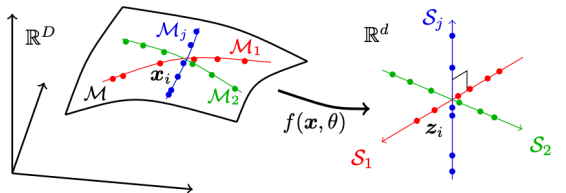
Tokens \rightarrow *embeddings* $\mathbf{X} = [\mathbf{x}^1, \dots, \mathbf{x}^N] \in \mathbb{R}^{D \times N}$.



A Geometric View of Coding Rate Reduction

Recall the *supervised* rate reduction:

$$\Delta R(\mathbf{Z} \mid \Pi) := R(\mathbf{Z}) - \underbrace{\sum_{k=1}^K \frac{n_k}{n} R(\mathbf{Z}_k)}_{R^c(\mathbf{Z} \mid \Pi)}.$$

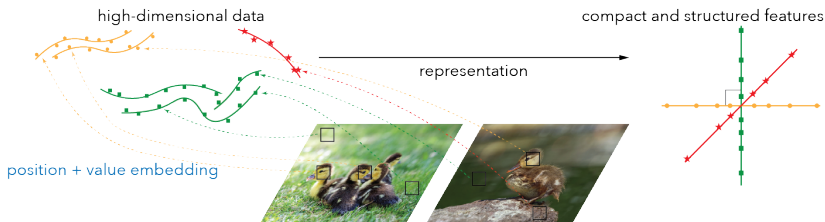


Rate Reduction for Token Sequences

Rate reduction for **sequence data**:

Parameterize the GMM covariances $\Sigma_k = U_k U_k^\top$.

$$\Delta R(\underbrace{\mathbf{Z} \mid \mathbf{U}_{[K]}}_{:= (\mathbf{U}_k)_{k=1}^K}) := R(\mathbf{Z}) - \underbrace{\sum_{k=1}^K R(\mathbf{U}_k^\top \mathbf{Z})}_{:= R^c(\mathbf{Z} | \mathbf{U}_{[K]})}$$

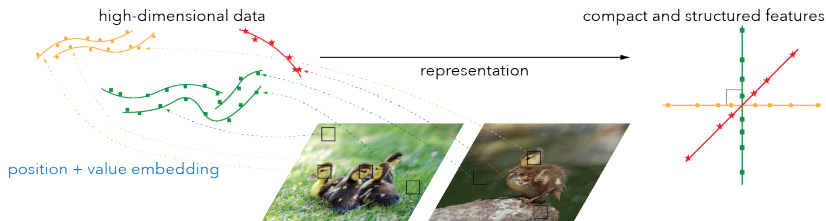


Sparse Rate Reduction

To be **maximally structured**, ask \mathbf{Z} (hence \mathbf{U}_k) to be **sparse**!

Objective to maximize: **Sparse Rate Reduction**

$$\text{SRR}(\mathbf{Z} \mid \mathbf{U}_{[K]}) := R(\mathbf{Z}) - R^c(\mathbf{Z} \mid \mathbf{U}_{[K]}) - \lambda \|\mathbf{Z}\|_1$$



Unrolling the Sparse Rate Reduction

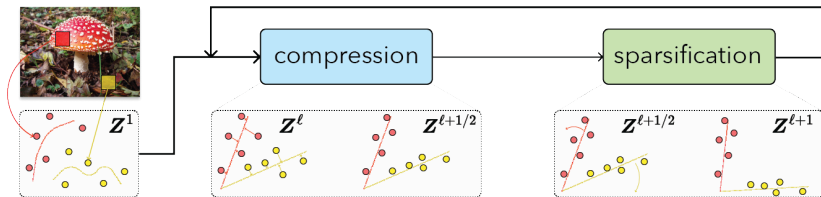
Proposed optimization strategy:

Two-step (prox-like) iteration.

$$\mathbf{Z}^\ell \mapsto \mathbf{Z}^{\ell+1/2} \mapsto \mathbf{Z}^{\ell+1}$$

$$\mathbf{Z}^{\ell+1/2} \approx \mathbf{Z}^\ell - \kappa \nabla_{\mathbf{Z}} R^c(\mathbf{Z}^\ell \mid \mathbf{U}_{[K]}^\ell) \quad (\text{compression})$$

$$\mathbf{Z}^{\ell+1} \approx \arg \max_{\mathbf{Z}: \mathbf{Z}^{\ell+1/2} = \mathbf{D}^\ell \mathbf{Z}} \{R(\mathbf{Z}) - \lambda \|\mathbf{Z}\|_1\} \quad (\text{sparsification})$$



Parameters: $(\mathbf{U}_k^\ell)_{k=1}^K \subseteq \mathbb{R}^{d \times p}$, $\mathbf{D}^\ell \in \mathbb{R}^{d \times d}$.

Gradient of Compression Objective

If $(U_k)_{k=1}^K \approx \text{orthogonal} + \text{p/w} \approx \text{orthogonal} + \approx \text{support } Z$:

$$\begin{aligned}\nabla_Z R^c(Z \mid U_{[K]}) &= \sum_{k=1}^K \beta (U_k U_k^\top Z) (I + \beta (U_k^\top Z)^\top (U_k^\top Z))^{-1} \\ &\approx \sum_{k=1}^K \beta U_k (U_k^\top Z) (I - \beta (U_k^\top Z)^\top (U_k^\top Z)) \\ &= \beta \left[\left(\sum_{k=1}^K U_k U_k^\top \right) Z - \beta \sum_{k=1}^K U_k (U_k^\top Z) (U_k^\top Z)^\top (U_k^\top Z) \right] \\ &\approx \beta \left[Z - \beta \sum_{k=1}^K U_k (U_k^\top Z) (U_k^\top Z)^\top (U_k^\top Z) \right]\end{aligned}$$

Gradient shaping/"non-parametric autoregression":

$$\nabla_Z R^c(Z) \approx \beta \left[Z - \beta \sum_{k=1}^K U_k (U_k^\top Z) \text{softmax} \left\{ (U_k^\top Z)^\top (U_k^\top Z) \right\} \right]$$

Multi-head Subspace Self-Attention

$$\nabla_{\mathbf{Z}} R^c(\mathbf{Z}) \approx \beta \left[\mathbf{Z} - \beta \sum_{k=1}^K \mathbf{U}_k (\mathbf{U}_k^\top \mathbf{Z}) \text{softmax} \left\{ (\mathbf{U}_k^\top \mathbf{Z})^\top (\mathbf{U}_k^\top \mathbf{Z}) \right\} \right]$$

Multi-head Subspace Self-Attention (MSSA):

$$\text{MSSA}(\mathbf{Z} \mid \mathbf{U}_{[K]}) := \beta [\mathbf{U}_1, \dots, \mathbf{U}_K] \begin{bmatrix} (\mathbf{U}_1^\top \mathbf{Z}) \text{softmax}\{(\mathbf{U}_1^\top \mathbf{Z})^\top (\mathbf{U}_1^\top \mathbf{Z})\} \\ \vdots \\ (\mathbf{U}_K^\top \mathbf{Z}) \text{softmax}\{(\mathbf{U}_K^\top \mathbf{Z})^\top (\mathbf{U}_K^\top \mathbf{Z})\} \end{bmatrix}$$

$$\mathbf{Z}^{\ell+1/2} := \underbrace{(1 - \beta\kappa) \mathbf{Z}^\ell}_{\text{residual}} + \underbrace{\beta\kappa \text{MSSA}(\mathbf{Z}^\ell \mid \mathbf{U}_{[K]})}_{\text{attention-like}}$$

Iterative Shrinkage-Thresholding Block

If $D^\ell \approx$ complete incoherent dictionary then

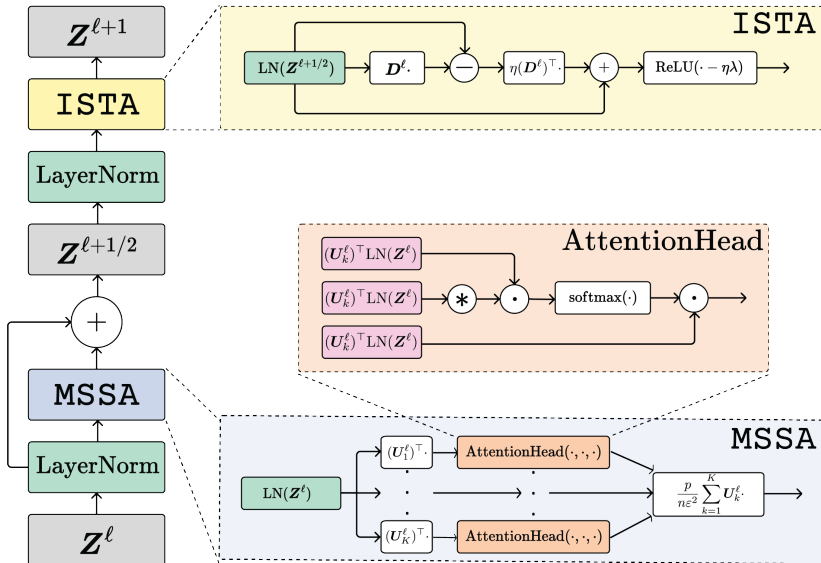
$$\mathbf{Z}^{\ell+1/2} \approx D^\ell \mathbf{Z} \implies R(\mathbf{Z}) \approx R(\mathbf{Z}^{\ell+1/2})$$

Can simplify the prox-like step:

$$\begin{aligned} \mathbf{Z}^{\ell+1} &\approx \arg \max_{\mathbf{Z}: \mathbf{Z}^{\ell+1/2} \approx D^\ell \mathbf{Z}} \{R(\mathbf{Z}) - \lambda \|\mathbf{Z}\|_1\} \approx \arg \min_{\mathbf{Z}: \mathbf{Z}^{\ell+1/2} \approx D^\ell \mathbf{Z}} \|\mathbf{Z}\|_1 \\ &\approx \arg \min_{\mathbf{Z}} \left\{ \frac{1}{2} \|\mathbf{Z}^{\ell+1/2} - D^\ell \mathbf{Z}\|_2^2 + \lambda' \|\mathbf{Z}\|_1 \right\} \end{aligned}$$

$$\begin{aligned} \mathbf{Z}^{\ell+1} &:= \text{ISTA}(\mathbf{Z}^{\ell+1/2}) \\ &:= \text{ReLU}(\mathbf{Z}^{\ell+1/2} + \lambda' D^{\ell\top} (\mathbf{Z}^{\ell+1/2} - D^\ell \mathbf{Z}^{\ell+1/2}) - \kappa \lambda' \mathbf{1}) \end{aligned}$$

CRATE Architecture

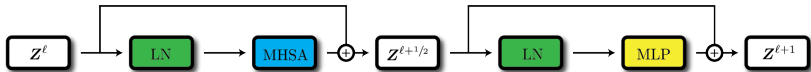


Comparing CRATE and Regular Transformer



Three practical differences:

- MSSA sets $W_{\text{query},k} = W_{\text{key},k} = W_{\text{value},k} = U_k^\top$
- ISTA sets $W_{\text{up}} = W_{\text{down}}^\top = D$
- In ISTA the residual connection is moved inside ReLU



Do CRATE Models Behave According to Theory?

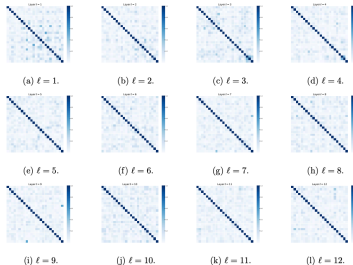
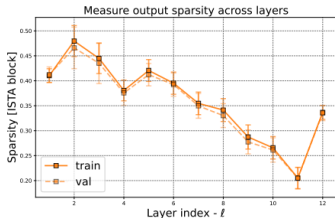
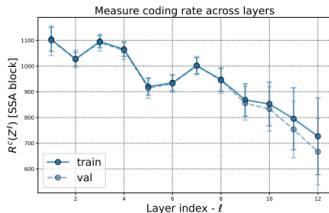


Figure 17: We visualize the $[U_1^\ell, \dots, U_K^\ell]^\top [U_1^\ell, \dots, U_K^\ell] \in \mathbb{R}^{pK \times pK}$ at different layers. The (i, j) -th

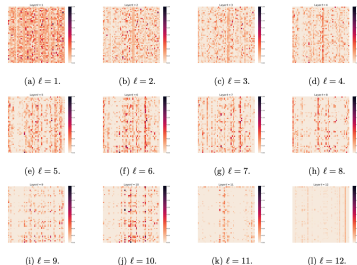


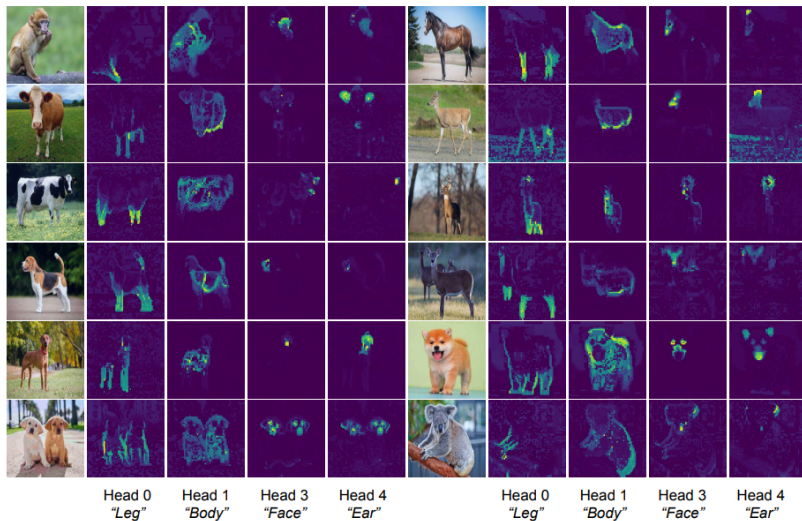
Figure 16: Visualizing layer-wise token Z^ℓ representations at each layer ℓ . To enhance the visual

Can CRATE Perform Well in Practice?

Model	CRATE-T	CRATE-S	CRATE-B	CRATE-L	ViT-T	ViT-S
# parameters	6.09M	13.12M	22.80M	77.64M	5.72M	22.05M
ImageNet-1K	66.7	69.2	70.8	71.3	71.5	72.4
ImageNet-1K ReaL	74.0	76.0	76.5	77.4	78.3	78.4
CIFAR10	95.5	96.0	96.8	97.2	96.6	97.2
CIFAR100	78.9	81.0	82.7	83.6	81.8	83.2
Oxford Flowers-102	84.6	87.1	88.7	88.3	85.1	88.5
Oxford-IIIT-Pets	81.4	84.9	85.3	87.4	88.5	88.6



Interpretability and Emergent Segmentation



Summary

Main takeaway: We can use unrolled optimization with representation learning objectives to build white-box deep networks which perform well and efficiently at scale!

- Seen some examples of how to \approx recover existing architectures: ReduNet (CNN), CRATE (Transformer)
- **Next lecture:**
 - Different unrolling strategies, different optimization algorithms, different objectives \implies *different, novel* architectures with favorable empirical properties!
 - Warmup: Using the framework to obtain white-box LLMs.

References I

- [Car+21] Mathilde Caron, Hugo Touvron, Ishan Misra, et al. “Emerging properties in self-supervised vision transformers”. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 9650–9660.
- [Cha+22] Kwan Ho Ryan Chan, Yaodong Yu, Chong You, et al. “ReduNet: A white-box deep network from the principle of maximizing rate reduction”. In: *Journal of machine learning research* 23.114 (2022), pp. 1–103.

References II

- [Dev+19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, et al. “Bert: Pre-training of deep bidirectional transformers for language understanding”. In: *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*. 2019, pp. 4171–4186.
- [Dos+20] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, et al. “An image is worth 16x16 words: Transformers for image recognition at scale”. In: *arXiv preprint arXiv:2010.11929* (2020).
- [GL10] Karol Gregor and Yann LeCun. “Learning fast approximations of sparse coding”. In: *Proceedings of the 27th international conference on international conference on machine learning*. 2010, pp. 399–406.

References III

- [He+16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, et al. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.
- [Rad+19] Alec Radford, Jeffrey Wu, Rewon Child, et al. “Language models are unsupervised multitask learners”. In: *OpenAI blog* 1.8 (2019), p. 9.
- [Rad+23] Ilija Radosavovic, Tete Xiao, Stephen James, et al. “Real-world robot learning with masked visual pre-training”. In: *Conference on Robot Learning*. PMLR. 2023, pp. 416–426.
- [Vas+17] Ashish Vaswani, Noam Shazeer, Niki Parmar, et al. “Attention is all you need”. In: *Advances in neural information processing systems* 30 (2017).

References IV

- [Wan+24] Peng Wang, Huikang Liu, Druv Pai, et al. “A global geometric analysis of maximal coding rate reduction”. In: *arXiv preprint arXiv:2406.01909* (2024).
- [Xie+17] Saining Xie, Ross Girshick, Piotr Dollár, et al. “Aggregated residual transformations for deep neural networks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1492–1500.
- [Yu+20] Yaodong Yu, Kwan Ho Ryan Chan, Chong You, et al. “Learning diverse and discriminative representations via the principle of maximal coding rate reduction”. In: *Advances in neural information processing systems* 33 (2020), pp. 9422–9434.

References V

- [Yu+23] Yaodong Yu, Sam Buchanan, Druv Pai, et al.
“White-box transformers via sparse rate reduction”.
In: *Advances in Neural Information Processing Systems*
36 (2023), pp. 9422–9457.