

Representation geometry in task-optimized network models

Bin Wang

Advance theory course

02/05/2025

Outline

- Introduction

1. Neural representation
2. Task-optimized NNs (and its relevance to neuroscience)

- Empirical phenomena

1. Platonic representation hypothesis
2. Dimensionality expansion-compression
3. Neural collapse and abstract representation

- Theory

1. Linear NNs
2. Two-layer ReLU NNs

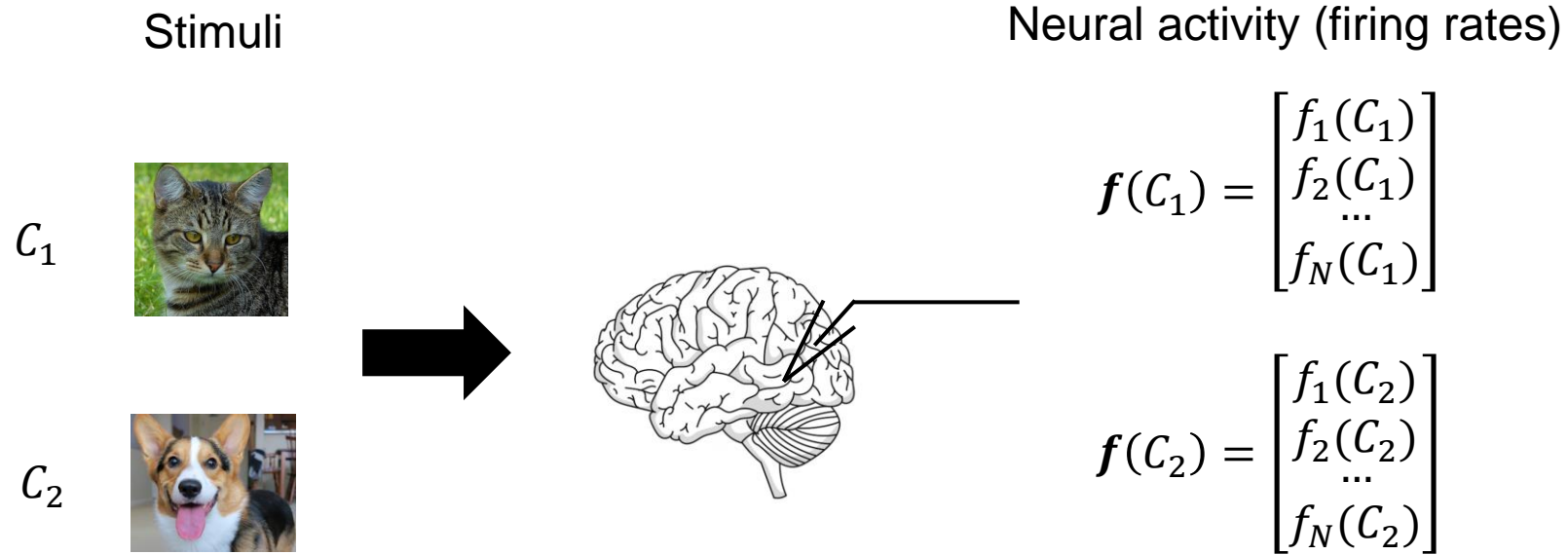
Outline

- **Introduction**

1. Neural representation

2. Task-optimized NNs

Neural representation: stimulus conditions

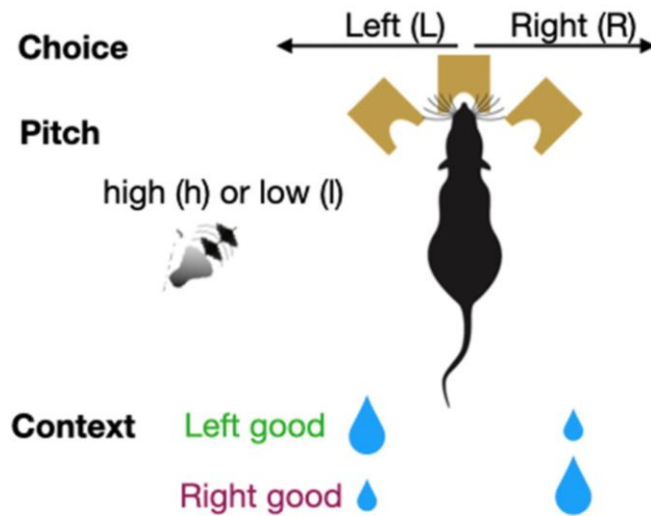


Neural activities in all stimulus conditions are encoded in a matrix:

		Condition			
Neuron		$f_1(C_1)$	$f_1(C_2)$	\dots	$f_1(C_p)$
		$f_2(C_1)$	$f_2(C_2)$	\dots	$f_2(C_p)$
		\dots	\dots	\dots	\dots
		$f_N(C_1)$	$f_N(C_2)$	\dots	$f_N(C_p)$
		Columns			

Rows

Neural representation: task conditions

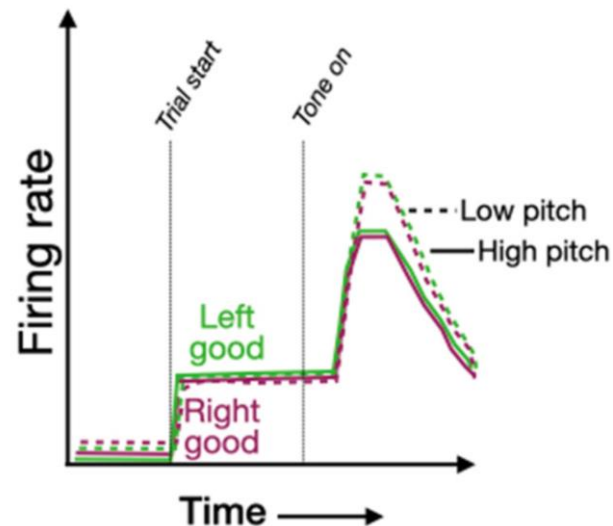


Each trial is uniquely associated with a “task condition”

$$C = (h, L, \emptyset)$$

Pitch Action Reward or not

In total, 8 distinct task conditions (3 binary variables).



Neural activity

$$f(C) = \begin{bmatrix} f_1(C) \\ f_2(C) \\ \dots \\ f_N(C) \end{bmatrix}$$

Neural
response
matrix

Condition				
Neuron	$f_1(C_1)$	$f_1(C_2)$...	$f_1(C_p)$
	$f_2(C_1)$	$f_2(C_2)$...	$f_2(C_p)$

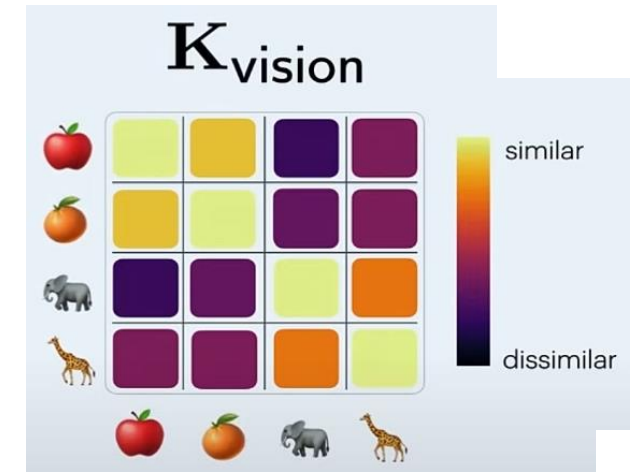
	$f_N(C_1)$	$f_N(C_2)$...	$f_N(C_p)$
	Columns			
Rows				

Row and column space of the representation matrix

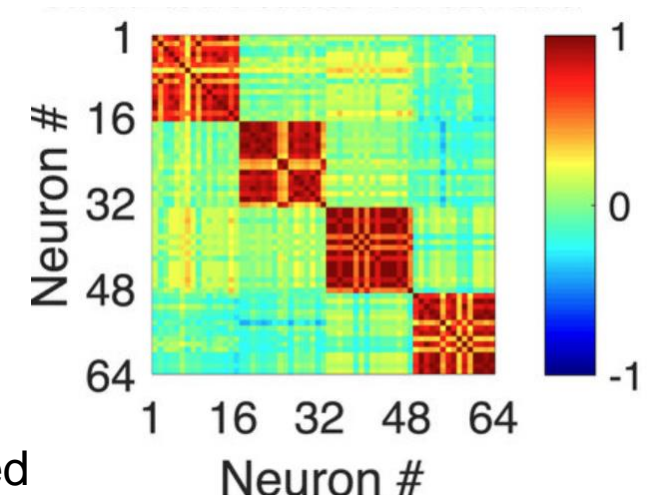
$$\mathbf{R} = \begin{matrix} & \text{Condition} & & \\ \text{Neuron} & \begin{pmatrix} f_1(C_1) & f_1(C_2) & \dots & f_1(C_p) \\ f_2(C_1) & f_2(C_2) & \dots & f_2(C_p) \\ \vdots & \vdots & \ddots & \vdots \\ f_N(C_1) & f_N(C_2) & \dots & f_N(C_p) \end{pmatrix} & \text{Rows} \\ & \text{Columns} & & \end{matrix}$$

$$K = R^T R$$

$$K_n = R R^T$$

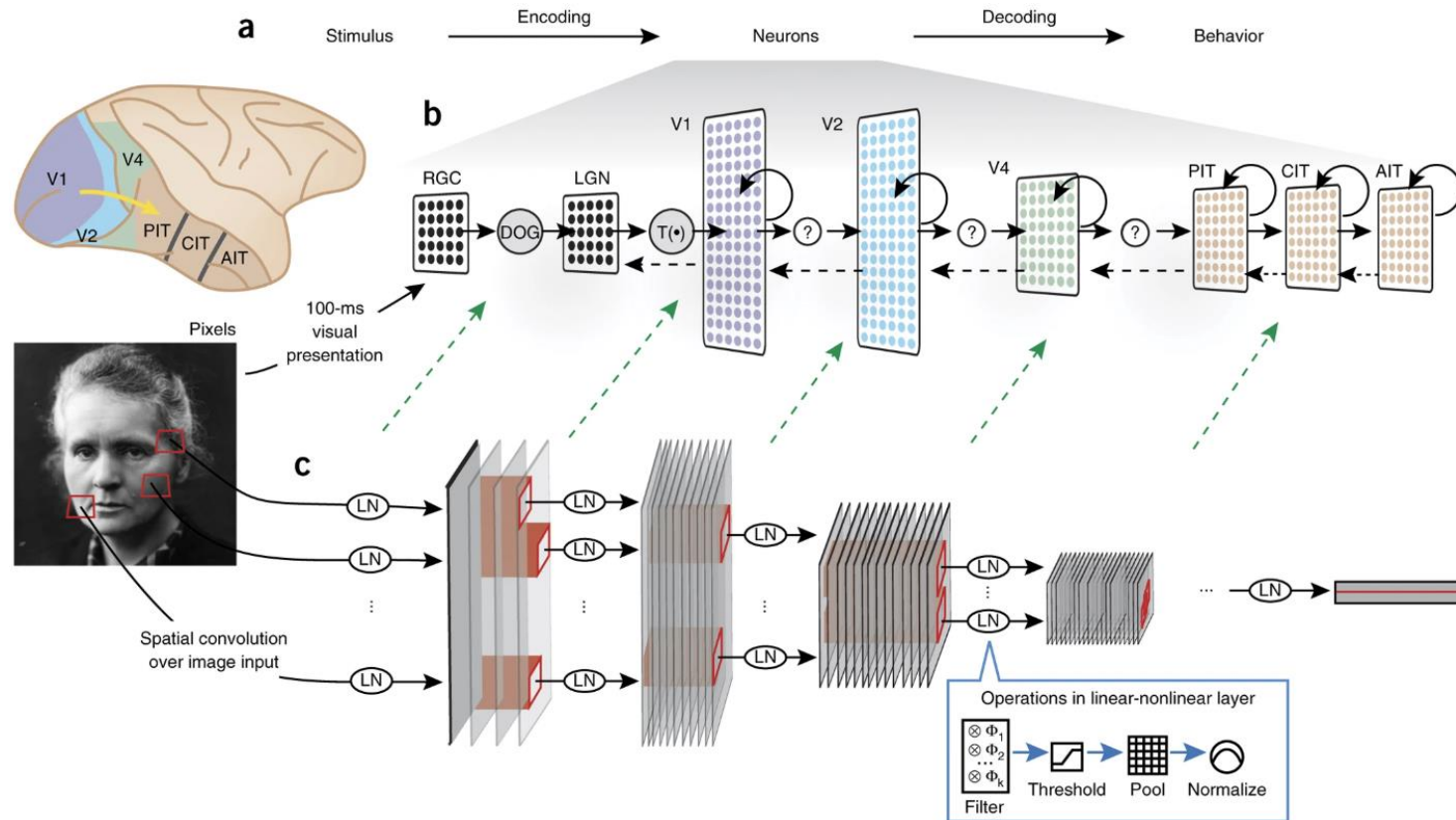


“Representation kernel”



Representation geometry is the geometry defined by the representation kernel matrix.

Task-optimized neural network models: sensory region



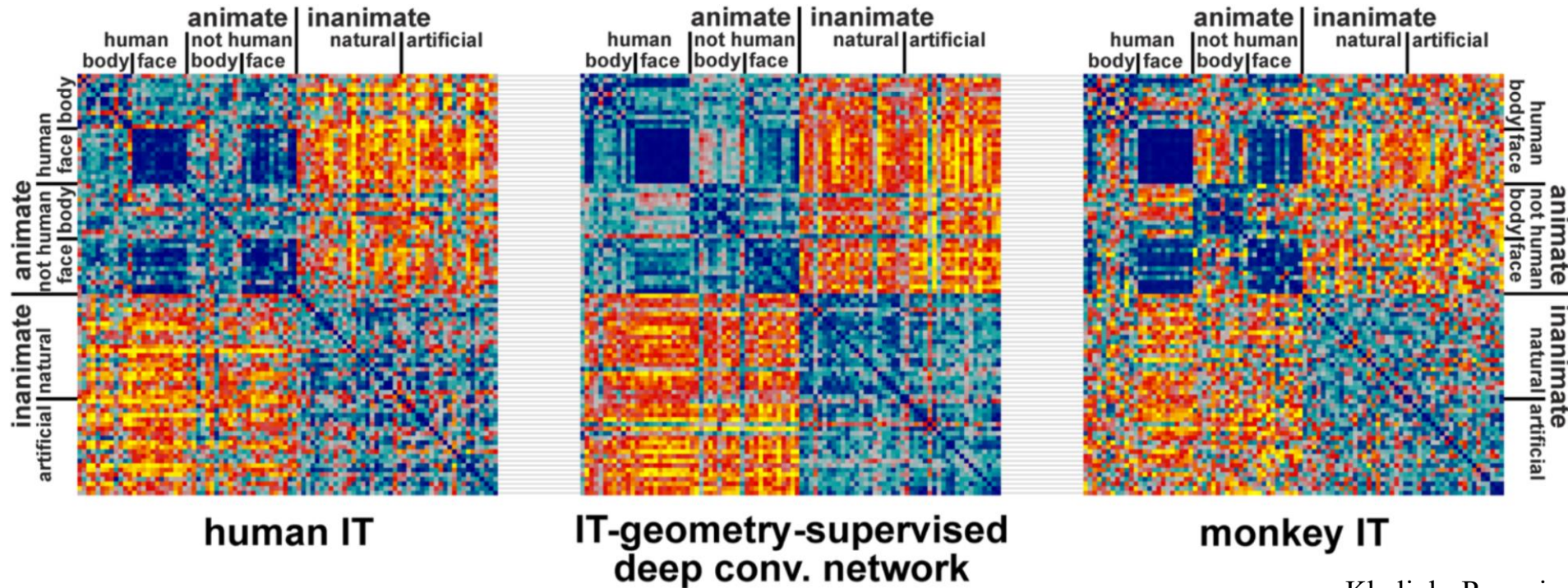
(See Tahereh Toosi's lecture)

Optimize the loss function via gradient-based method

$$L(W) = \sum_{i=1}^P l(y_i, f_W(x_i)) + R(W) ,$$

where f_W is a deep neural network.

Neural representation in models vs. in the brain



(See Tahereh Toosi's lecture)

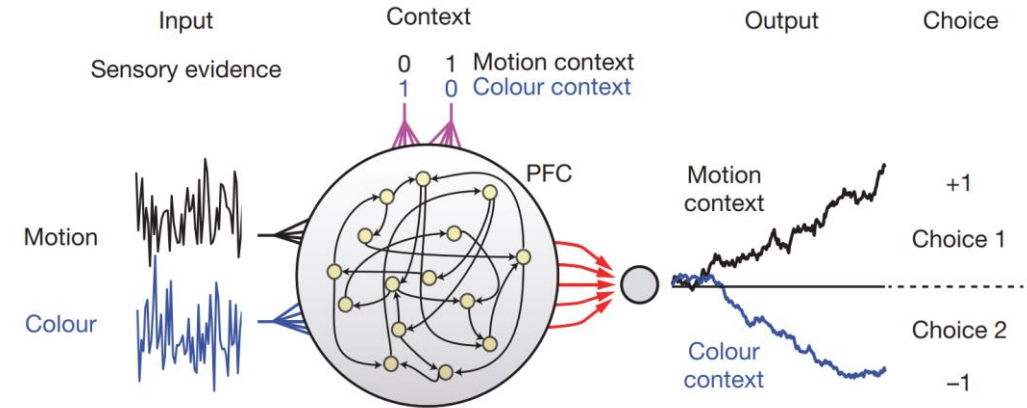
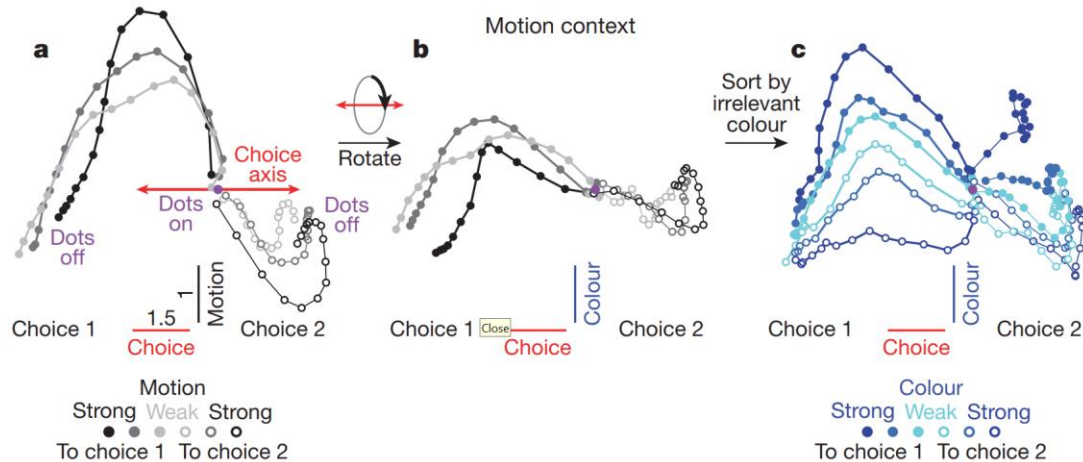
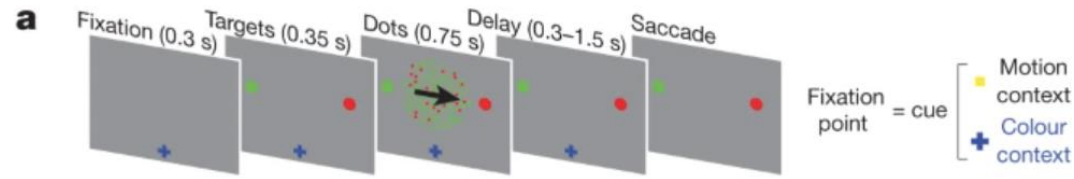
Khaligh-Razavi and Kriegeskorte.
PLoS Comp. Biol. (2014)

Task- optimized network also shows similar neural representations in other brain regions, like V4 etc.

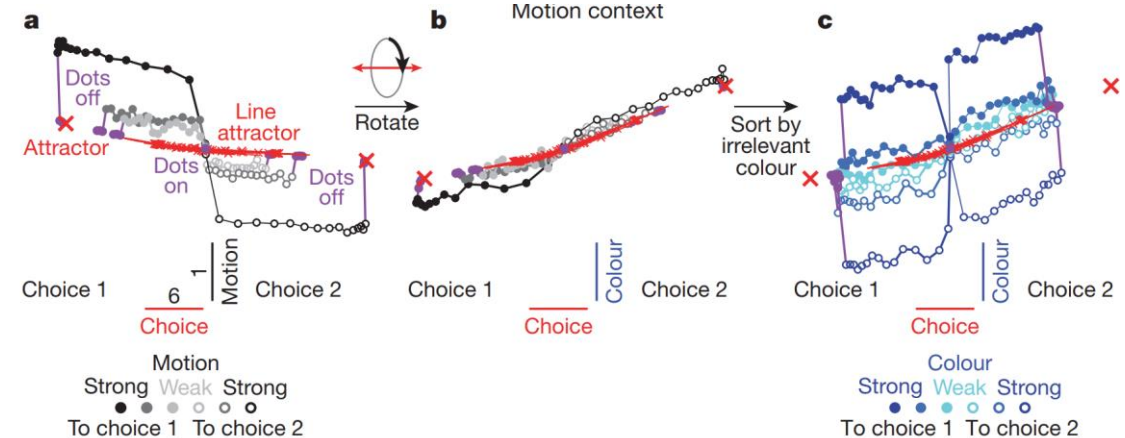
Task-optimized neural network models: cognitive region

Context-dependent computation by recurrent dynamics in prefrontal cortex

Valerio Mante^{1,*,}, David Sussillo^{2,*,}, Krishna V. Shenoy^{2,3} & William T. Newsome¹



(See Tahereh Toosi's lecture)



Task-optimized RNNs generate similar neural population dynamics as in the data.

And many others

The quest to identify structures in neural representation

Main question:

What neural representation does a task-optimized network model develop for a given set of tasks?

Sub-questions:

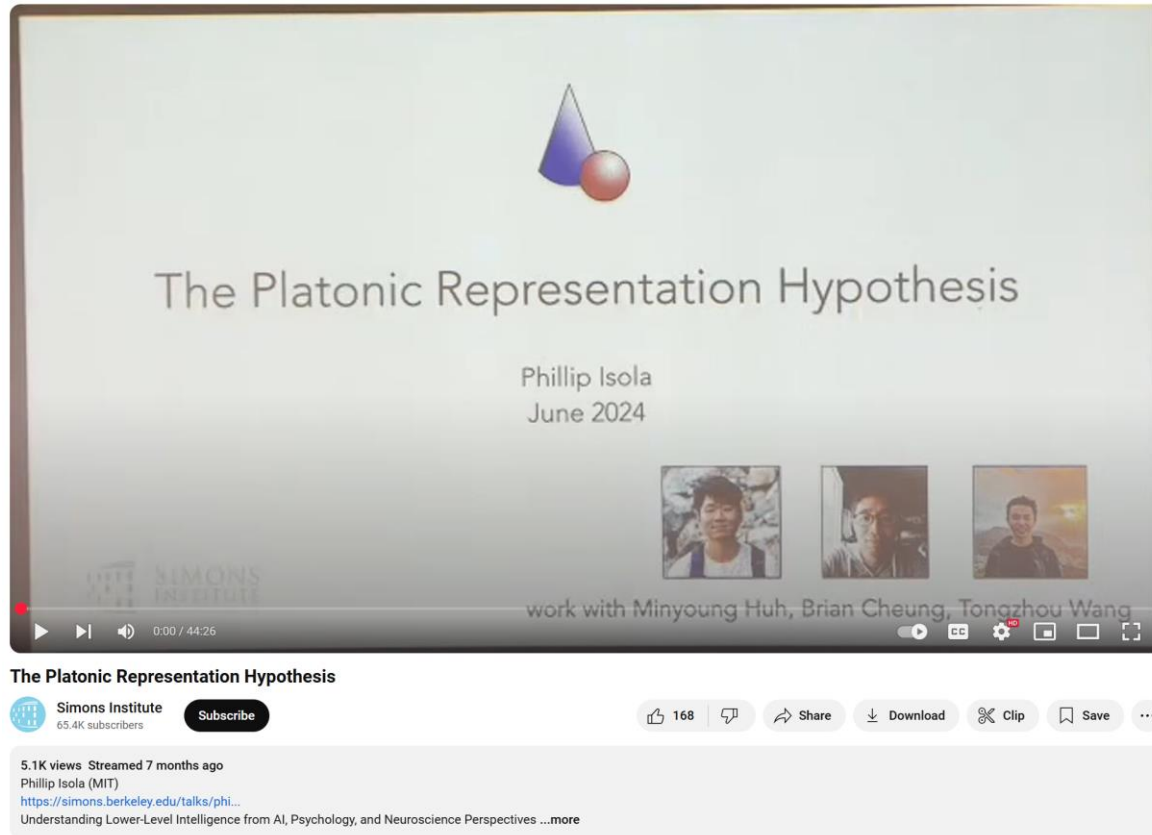
1. Will the optimization procedure always find a unique neural representation?
2. Beyond the task structure, does the neural representation depend on other properties of the model (size, architecture, regularization)?
3. Shared structure vs. individual variability in neural representations

....

Outline

- **Empirical phenomena**
 1. **Platonic representation hypothesis**
 2. Dimensionality expansion-compression
 3. Neural collapse and abstract representation

Platonic representation hypothesis



Short summary:

Different neural network models, trained with different objectives on naturalistic data from different modalities, are converging to similar representations.

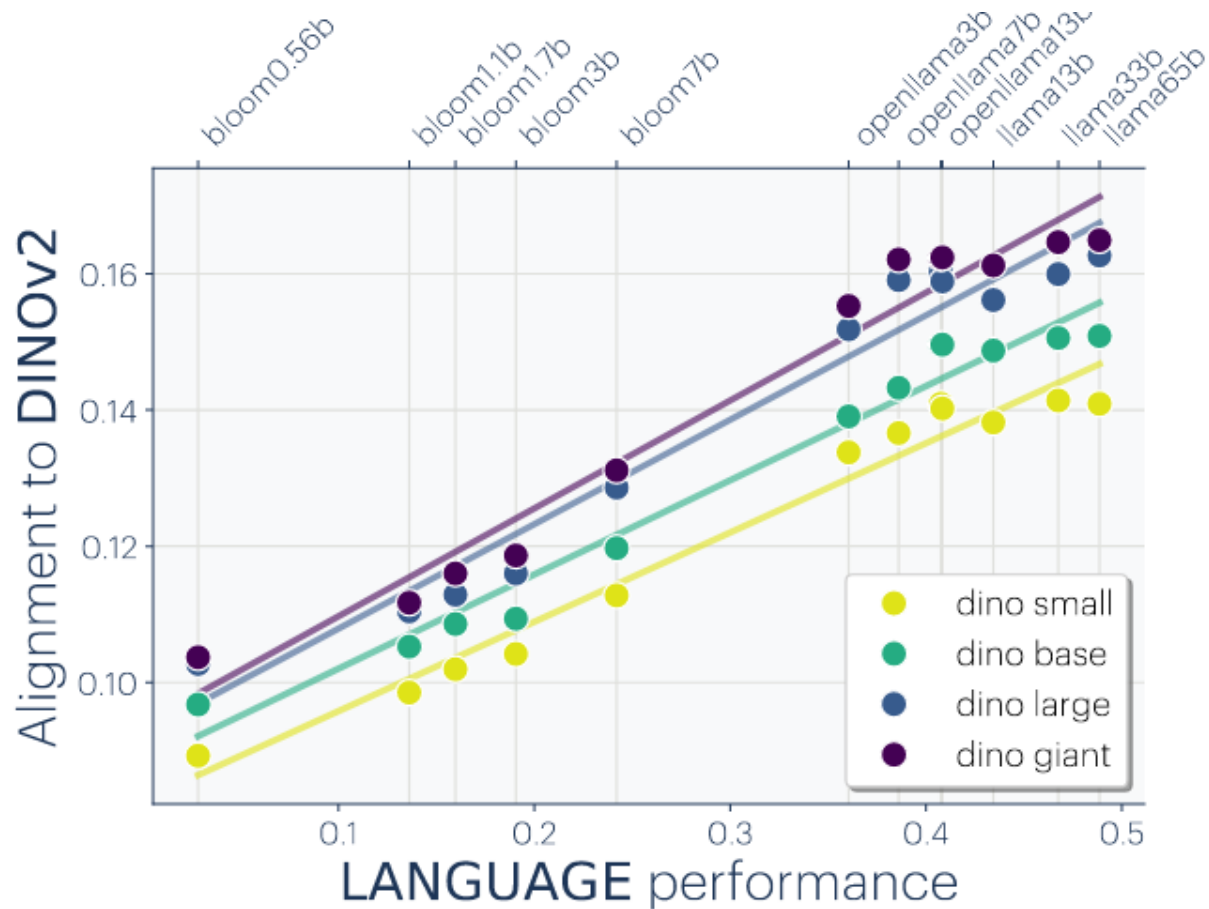
(if they perform well on downstream tasks)

Huh et al. ICLR (2024)

Similar phenomena were also observed in previous works.
(hidden manifold hypothesis etc.)

Language models with better performance have more aligned representation to visual models

Alignment
between
representation
kernels



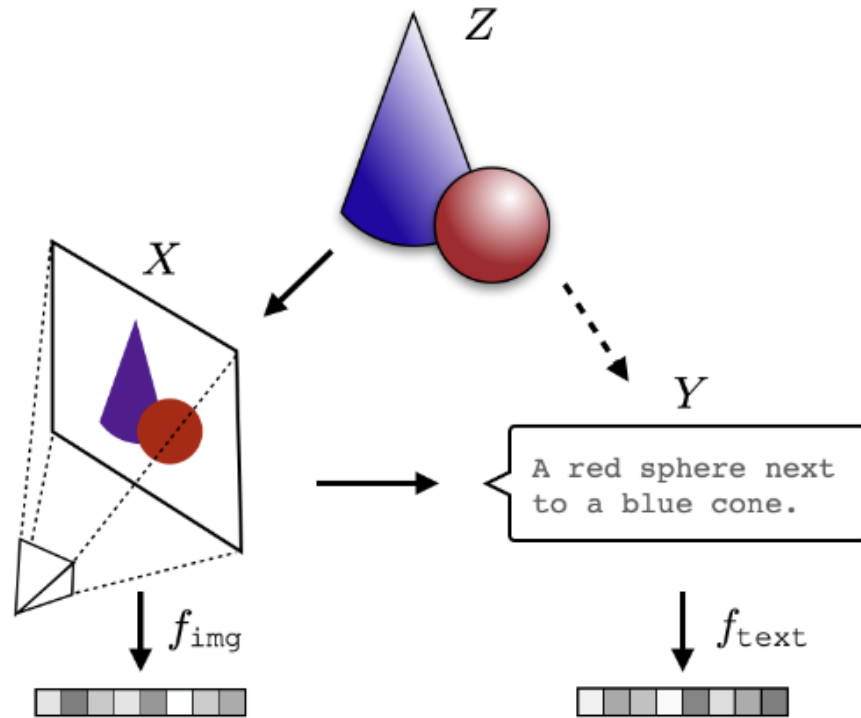
Similar trends exist for
different visual models.

Compare: representation of an image vs. representation of its caption

Shared “latent structure” in the naturalistic data

Similar neural representations are developed for

1. different model architectures;
2. different data modalities



Q:

1. What is the underlying latent structure?
2. What is this shared representation? How to characterize it?

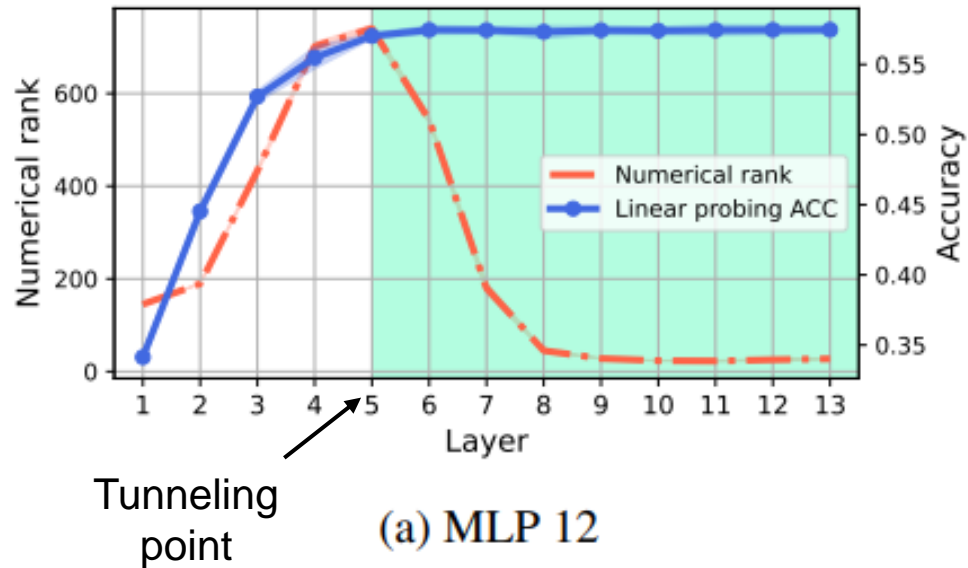
Outline

- **Empirical phenomena**

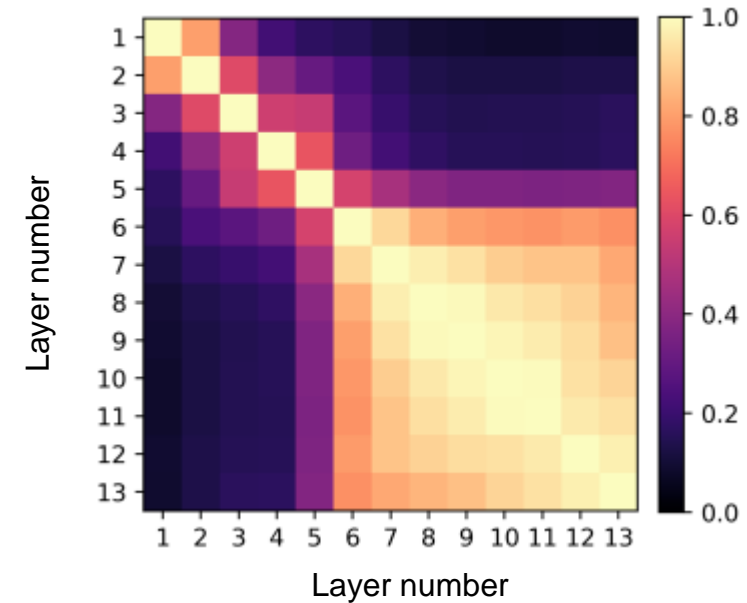
1. Platonic representation hypothesis
- 2. Dimensionality expansion-compression**
3. Neural collapse and abstract representation

Large-scale properties of neural representation across layers

CIFAR-10

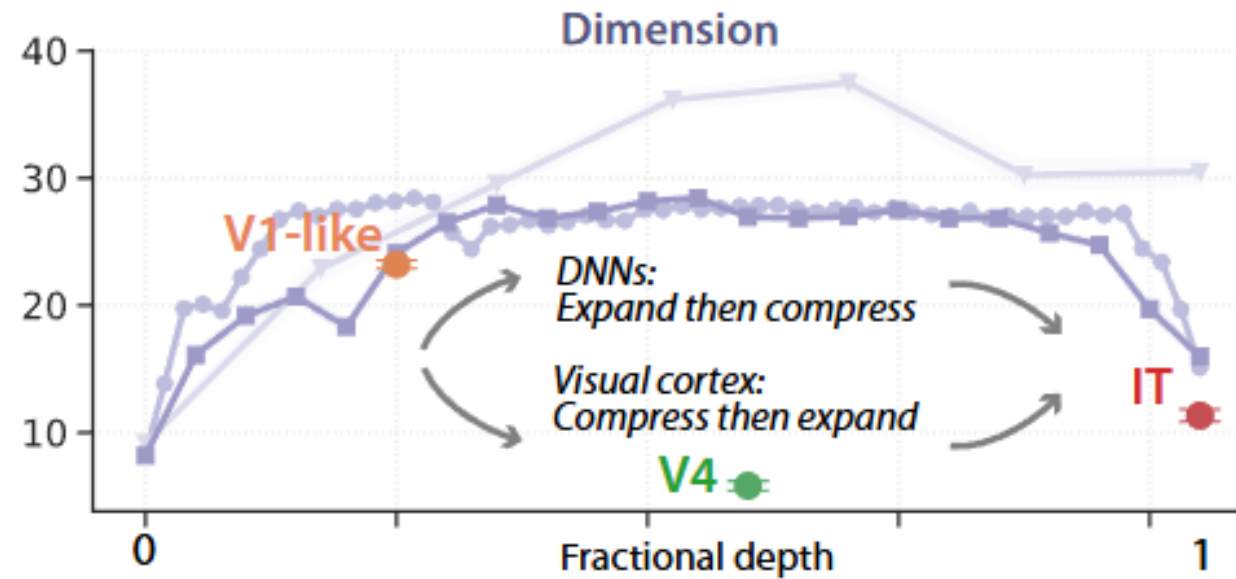


Similarity between representation kernels across different layers



- Replacing layers after the “tunneling point” by linear layers doesn’t change performance
- Robust across different model architecture

Difference in the dimensionality along visual hierarchy



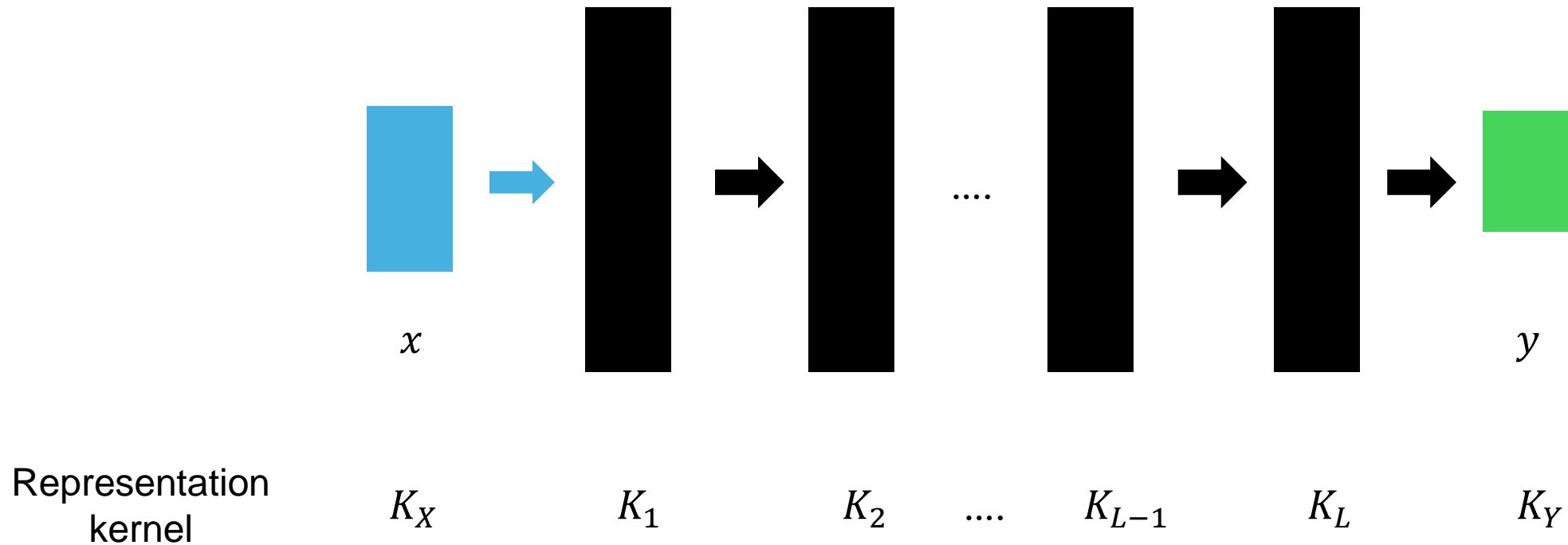
Sorscher, Ganguli and Sompolinsky, *PNAS* (2022)

Outline

- **Empirical phenomena**
 1. Platonic representation hypothesis
 2. Dimensionality expansion-compression
 - 3. Neural collapse and abstract representation**

The general phenomena of (incomplete) kernel alignment

Feedforward network



Expect: $K_L \approx K_Y + (\text{corrections})$.

Will it always happen? If not, when?

Neural collapse: one-hot output label

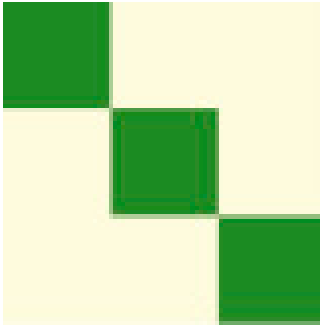
One-hot class labels (3 classes):

$$y_{1,i} = (1,0,0)$$

$$y_{2,i} = (0,1,0)$$

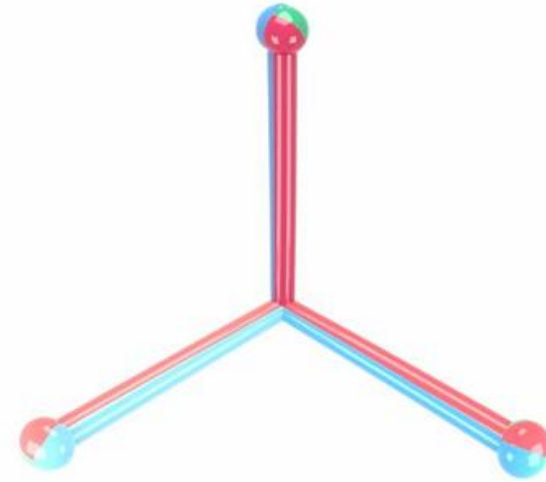
$$y_{3,i} = (0,0,1)$$

K_Y



After training

$$K_L \propto K_Y$$



Equiangular tight frame (ETF)
for centered last-layer kernel

Seems to happen for different model architecture trained on different image datasets

Abstract representation: binary output labels

Binary class labels (8 classes):

$$y_{1,i} = (+1, +1, +1)$$

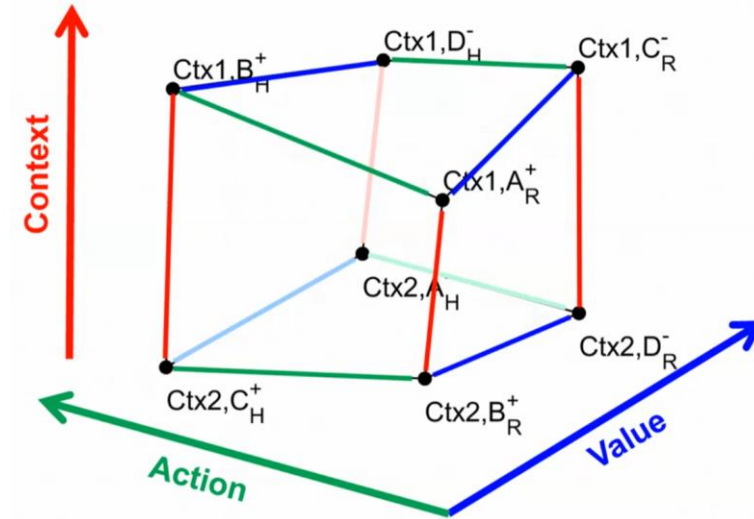
$$y_{2,i} = (-1, +1, +1)$$

...

$$y_{8,i} = (-1, -1, -1)$$

Output kernel: vertices of a hypercube

ACC—monkey ($K_L \propto K_Y$)



Similar representation geometry appears in other brain areas and in task-optimized networks

(See recent works in Fusi Lab)

But the story is not simple:

1. Collapse is not perfect
2. Input-dependent, architecture-dependent etc.

Outline

- **Theory**

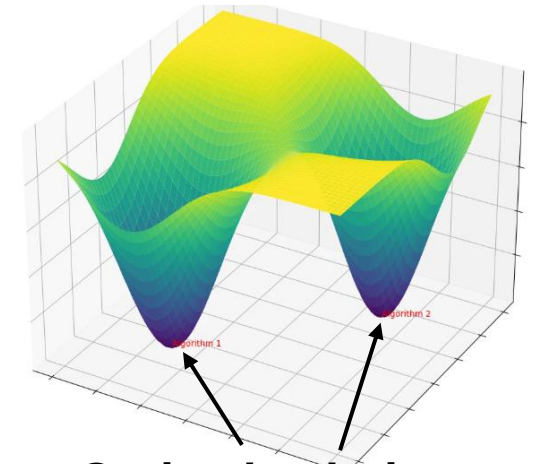
1. Linear NNs

2. Two-layer ReLU NNs

Overview of existing theoretical approaches

1. Loss landscape analysis

(Properties of local/global mins, saddles etc.
Structures of sublevel sets)



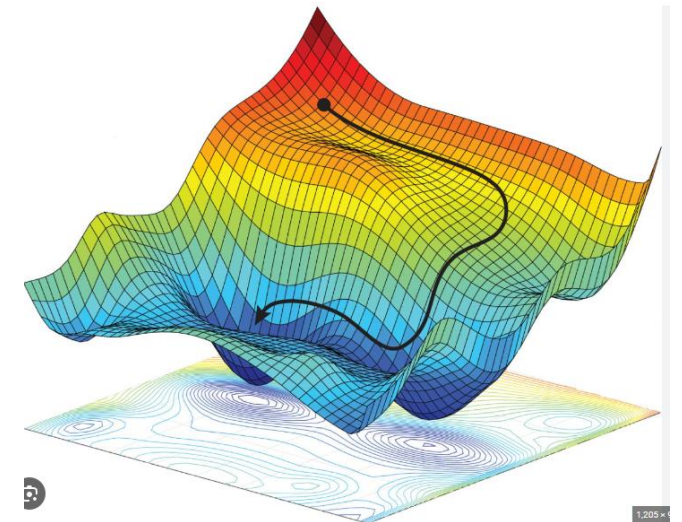
Optimal solution weights

2. Bayesian neural network

(Next lecture by Haozhe Shan)

3. Learning dynamics

(Trajectory-based analysis;
Depends on weight inti./algorithm;
Richer information but more difficult)



Outline

- **Theory**

1. **Linear NNs**

2. Two-layer ReLU NNs

Review of linear algebra

$$W \in \mathbb{R}^{M \times N},$$

singular values $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r > 0$.

Schatten-p norm:

$$\|W\|_p = \left(\sum_{i=1}^r \sigma_i^p \right)^{\frac{1}{p}}.$$

$p \geq 1$: $\|\cdot\|_p$ is a norm ($p = 2$, Frobenius norm). $0 < p < 1$: quasi-norm.

Singular value interlacing properties:

\bar{W} is the matrix after removing one row or column from W , then

$$\sigma_{i+1}(W) \leq \sigma_i(\bar{W}) \leq \sigma_i(W).$$

Corollary: Removing rows or columns always decreases Schatten-p norm.

Set-up for linear models

Input data: $X = (x_1, x_2, \dots, x_P) \in \mathbb{R}^{d_X \times P}$

Output labels: $Y = (y_1, y_2, \dots, y_P) \in \mathbb{R}^{d_Y \times P}$

Minimize

$$L(W) = ||Y - W_L W_{L-1} \dots W_1 X||_F^2 + \lambda \sum_{l=1}^L ||W_l||_F^2$$

All hidden layer sizes are $M \in \mathbb{N}_+$.

Unregularized case ($\lambda = 0$):

local minimum = global minimum

Kawaguchi, *NeurIPS* (2016)
Laurent and Brecht. *PMLR* (2018).

Will mainly focus on

wide hidden layer $M \gg 1$, small weight regularization $\lambda \rightarrow 0^+$.

Optimal solution for deep linear network

Theorem: If $M > \text{rank}(X), \text{rank}(Y)$, $\lambda \rightarrow 0^+$, then the optimal weights must satisfy

$$W \equiv W_L W_{L-1} \dots W_2 W_1 = Y X^\dagger,$$

where X^\dagger is the Moore-Penrose inverse of X .

Moreover, if $Y X^\dagger = U_{L+1} S U_1^T$ is the SVD decomposition, the optimal weights have the following forms

$$W_l = U_{l+1} \sqrt[l]{S} U_l^T, \quad l = 1, 2, \dots, L,$$

where U_L, \dots, U_2 are (any) orthogonal matrices.

Corollary: If input is whitened ($X = I_{d_X}$), then the l th-layer representation kernel is

$$K_l = (W_l \dots W_1 X)^T W_l \dots W_1 X = U_1 \sqrt[l]{S^{2l}} U_1^T.$$

In particular, if output is isotropic ($\sigma_i(Y) = \sigma$), then $K_l \propto K_Y$.

Proof

WLOG, assume that $Y \subseteq \text{Range}(X)$.

Lemma 1: Variational characterization for Schatten p-norm

$$||W||_{2/L}^{L/2} = \min \left\{ \sum_{l=1}^L ||W_l||_F^2 \mid W = W_L W_{L-1} \dots W_2 W_1 \right\}.$$

Equality if and only if $W_l = U_{l+1} D U_l^T$ for $l = 1, 2, \dots, L$, where $W = U_{L+1} D U_1$.

$$\text{Equivalent opt. problem: } ||Y - WX||_F^2 + \lambda ||W||_{2/L}^{L/2}.$$

Lemma 2: $\lambda \rightarrow 0^+$, the optimization problem is equivalent to

$$\min ||W||_{2/L}^{L/2},$$

subject to

$$Y = WX.$$

Use singular value interlacing property.

Outline

- **Theory**

1. Linear NNs

- 2. Two-layer ReLU NNs**

Set-up for two-layer ReLU network

Input data: $X = (x_1, x_2, \dots, x_P) \in \mathbb{R}^{d_X \times P}$

Output labels: $Y = (y_1, y_2, \dots, y_P) \in \mathbb{R}^{d_Y \times P}$

Minimize

$$L(W) = ||Y - W_2 \phi(W_1 X)||_F^2 + \lambda \sum_{l=1}^2 ||W_l||_F^2$$

Hidden layer size is $M \in \mathbb{N}_+$. ϕ is ReLU.

Unregularized case ($\lambda = 0$):

1. local minimum \neq global minimum
2. All global minima are connected for large M ($\phi \in \mathcal{C}^1$).

Simsek et al. *PMLR* (2021) and many previous works.

Will mainly focus on

wide hidden layer $M \gg 1$, small weight regularization $\lambda \rightarrow 0^+$.

Review of convex optimization

Any optimization problem

$$\begin{array}{ll} \min & f_0(x) \\ \text{s.t.} & f_i(x) \leq 0, \\ & h_i(x) = 0. \end{array} \quad \textbf{(Primal)}$$

Dual problem

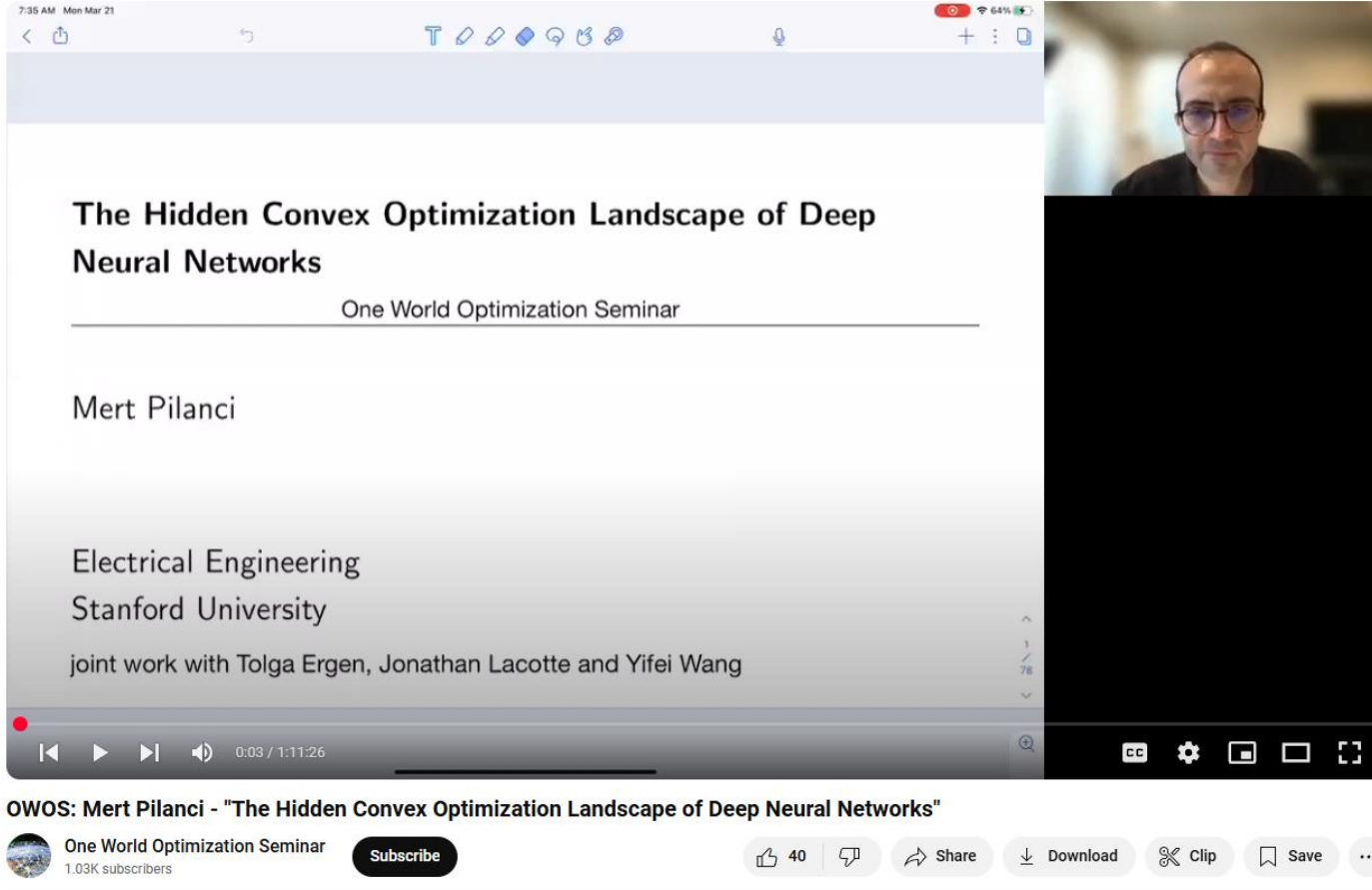
$$\max_{\lambda \geq 0, \mu} g(\lambda, \mu) \quad \textbf{(Dual)}$$

where $g(\lambda, \mu) = \inf_x L(x; \lambda, \mu) = \inf_x [f_0(x) - \sum_i \lambda_i f_i(x) - \sum_i \mu_i h_i(x)]$.

Two properties:

1. Dual problem is always a convex optimization problem (g is concave).
2. Bi-dual problem is a convex relaxation of the original problem.

Hidden convexity in two-layer ReLU network



The video player shows a presentation slide with the title "The Hidden Convex Optimization Landscape of Deep Neural Networks" and the subtitle "One World Optimization Seminar". The speaker is identified as Mert Pilanci, an Electrical Engineering student at Stanford University. The slide also mentions joint work with Tolga Ergen, Jonathan Lacotte, and Yifei Wang. The video player interface includes a progress bar at 0:03 / 1:11:26, a video thumbnail of the speaker, and a bottom bar with the video title, channel name "One World Optimization Seminar" (1.03K subscribers), and interaction buttons for likes (40), comments, share, download, clip, save, and a menu.

7:35 AM Mon Mar 21

The Hidden Convex Optimization Landscape of Deep Neural Networks

One World Optimization Seminar

Mert Pilanci

Electrical Engineering
Stanford University

joint work with Tolga Ergen, Jonathan Lacotte and Yifei Wang

0:03 / 1:11:26

OWOS: Mert Pilanci - "The Hidden Convex Optimization Landscape of Deep Neural Networks"

One World Optimization Seminar
1.03K subscribers

Subscribe

40

Share

Download

Clip

Save

See references from google scholar

Short summary:

For regularized two-layer ReLU NNs,

Bi-dual \approx Primal.

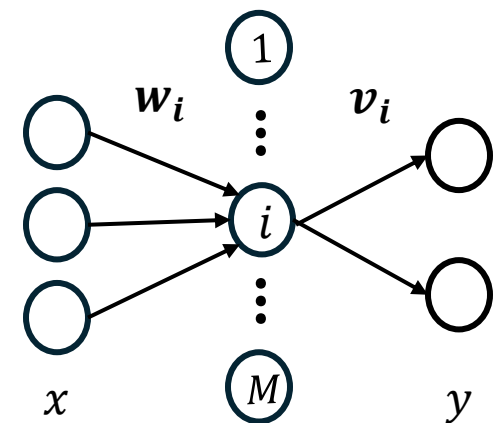
Only need to solve the **convex** bi-dual problem!

Can also generalize to deeper networks, attention layers etc.

The equivalent convex problem for two-layer ReLU NNs

ϕ is ReLU. $M > Pd_Y + 1$. Minimize

$$L(\{w_i, v_i\}) = \left\| Y - \sum_{i=1}^M v_i \phi(w_i^T X) \right\|_F^2 + \lambda \sum_{i=1}^M (\|w_i\|_2^2 + \|v_i\|_2^2).$$



Its equivalent convex problem (Constrained group LASSO)

$$L(\{V_i\}) = \left\| Y^T - \sum_{i=1}^p D_i X^T V_i \right\|_F^2 + \lambda \sum_{i=1}^p \|V_i\|_1,$$

where $D_i \in \mathcal{D} \in \mathbb{R}^{P \times P}$ encoding all possible sign patterns (total number is p)

$$\mathcal{D} = \{\text{diag}(\mathbf{1}_{X^T \mathbf{w} \geq 0}) \mid \|\mathbf{w}\|_2 \leq 1\}.$$

The domain of $V_i \in \mathbb{R}^{d_X \times d_Y}$ is

$$\mathcal{K}_i = \text{cone}\{\mathbf{w}\mathbf{v}^T \mid (2D_i - I)X^T \mathbf{w} \geq 0, \|\mathbf{v}\|_2 \leq 1\}.$$

Equivalence: Optimal solutions $V_i^* \rightarrow$ all optimal solutions of original problem

Spike-free data matrix and the bi-dual problem

The input matrix X is spike-free if

$$\{(X^T \mathbf{w})_+ \mid \|\mathbf{w}\|_2 \leq 1\} = \{X^T \mathbf{w} \mid \|\mathbf{w}\|_2 \leq 1\} \cap \mathbb{R}_{\geq 0}^P.$$

Lemma: Whitened input is spike-free.

The group LASSO simplifies for spike-free input

$$L(V) = \|Y^T - X^T V\|_F^2 + \lambda \|V\|_1$$

where the domain of V is $\mathcal{K} = \text{cone}\{\mathbf{w}\mathbf{v}^T \mid X^T \mathbf{w} \geq 0, \|\mathbf{v}\|_2 \leq 1\}$.

For any optimal solution $V^* = \sum_{j=1}^c \mathbf{a}_j \mathbf{b}_j^T$, $X^T \mathbf{a}_j \geq 0$, $\|\mathbf{b}_j\|_2 \leq 1$ and c is minimal, the normalized pair

$$(\mathbf{w}_j, \mathbf{v}_j) = \left(\sqrt{\frac{\|\mathbf{b}_j\|_2}{\|\mathbf{a}_j\|_2}} \mathbf{a}_j, \sqrt{\frac{\|\mathbf{a}_j\|_2}{\|\mathbf{b}_j\|_2}} \mathbf{b}_j \right), \quad j = 1, 2, \dots, c,$$

is optimal for original problem ($M \geq c$). And these are all the optimal solutions.

Optimal solution for whitened data

Corollary 1: If the input is whitened ($X = I_{d_X}$) and $Y^T = \sum_{i=1}^r \sigma_i a_i b_i^T$ is the SVD, which has the property that for each $\sigma_i > \lambda$, and its right singular vector satisfies $X^T b_i \geq 0$, then the optimal solution V^* is

$$V^* = \sum_{i=1}^r \left(\sigma_i - \frac{\lambda}{2} \right)_+ a_i b_i^T.$$

Proof: relax the original problem + Von Neumann trace ineqn.

Corollary 2: For whitened input ($X = I_{d_X}$), one-hot output labels with balanced class, $M > \text{rank}(Y)$ and small λ , the above optimal V^* corresponds to the neural collapse representation kernel,

$$K = \phi(W_1 X)^T \phi(W_1 X) \propto K_Y.$$

Not directly applicable to binary output labels and other nonlinearity.
(see another method on Monday lab meeting at Mar. 17th!)

Reference

- Framework to compute representation matrix

Kaufman, M. T., Benna, M. K., Rigotti, M., Stefanini, F., Fusi, S., & Churchland, A. K. The implications of categorical and category-free mixed selectivity on representational geometries. *Current opinion in neurobiology*, (2022).

- Planotic representation hypothesis

Huh, M., Cheung, B., Wang, T., & Isola, P. Position: The Platonic Representation Hypothesis. *ICLR* (2024)

- Tunnel effect and comparisons to primate visual hierarchy

Masarczyk, W., Ostaszewski, M., Imani, E., Pascanu, R., Miłoś, P., & Trzcinski, T. The tunnel effect: Building data representations in deep neural networks. *NeurIPS* (2024).

Sorscher, B., Ganguli, S., & Sompolinsky, H. Neural representational geometry underlies few-shot concept learning. *PNAS* (2022).

- Neural collapse and abstract representation

Papayan, Han, & Donoho. "Prevalence of neural collapse during the terminal phase of deep learning training." *PNAS* (2020).

Bernardi, S., Benna, M. K., Rigotti, M., Munuera, J., Fusi, S., & Salzman, C. D. The geometry of abstraction in the hippocampus and prefrontal cortex. *Cell* (2020)

- Deep linear network

Kawaguchi, Kenji. "On the theory of implicit deep learning: Global convergence with implicit layers." *arXiv* (2021).

Laurent, Thomas, and James Brecht. "Deep linear networks with arbitrary loss: All local minima are global." International conference on machine learning. *PMLR* (2018).

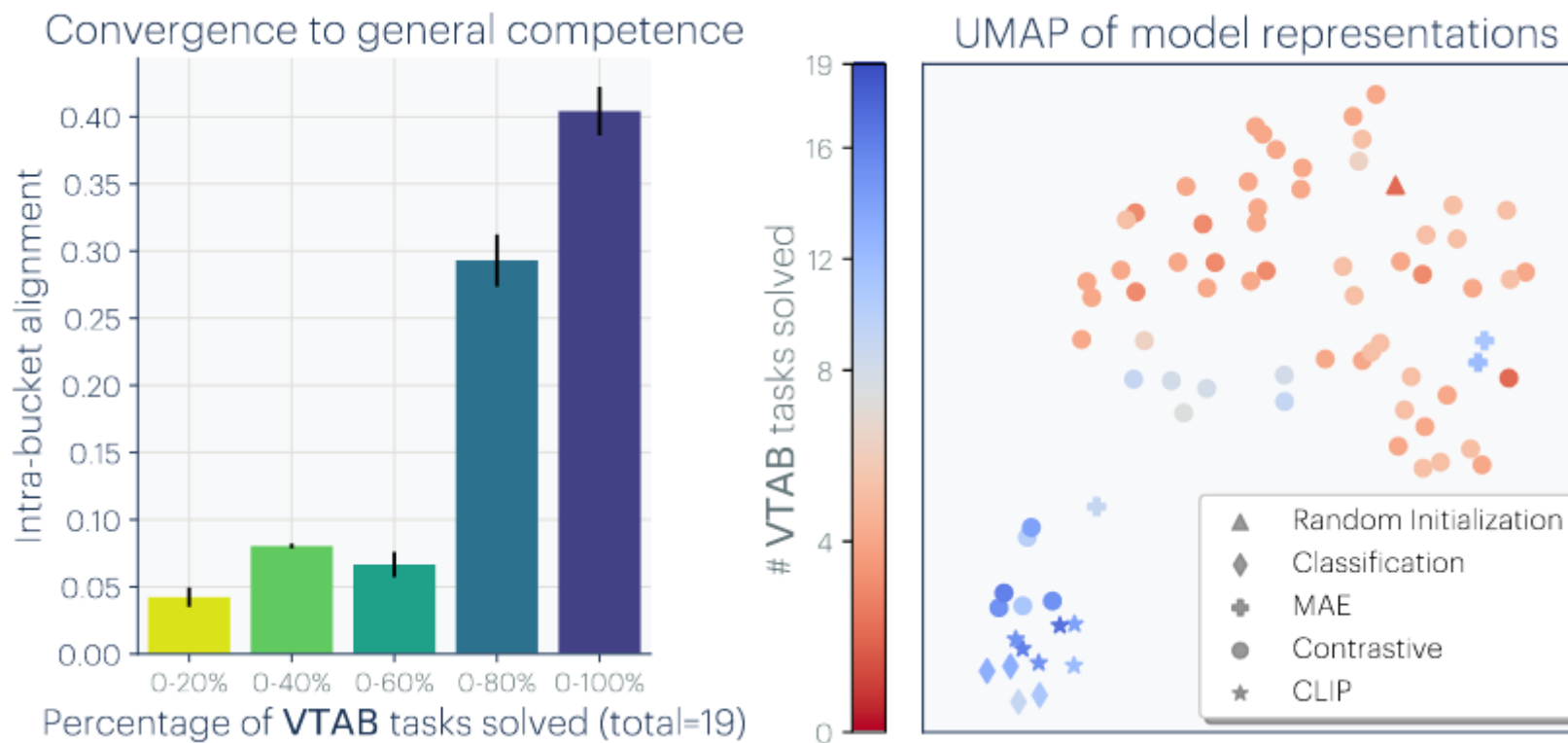
- Hidden convexity in two-layer ReLU network

Ergen, T., & Pilanci, M. Revealing the structure of deep neural networks via convex duality. *ICML* (2021).

Sahiner, A., Ergen, T., Pauly, J., & Pilanci, M. Vector-output relu neural network problems are copositive programs: Convex analysis of two-layer networks and polynomial-time algorithms. *ICLR* (2021).

Similar representations are formed in all “good” visual models

78 different vision models trained on different image datasets



- Alignment between representation kernels are measures via nearest-neighbor metrics
- VTAB = visual task adaptation benchmark