Understanding Neural Networks (2)

Speaker: Yi-Shan Wu

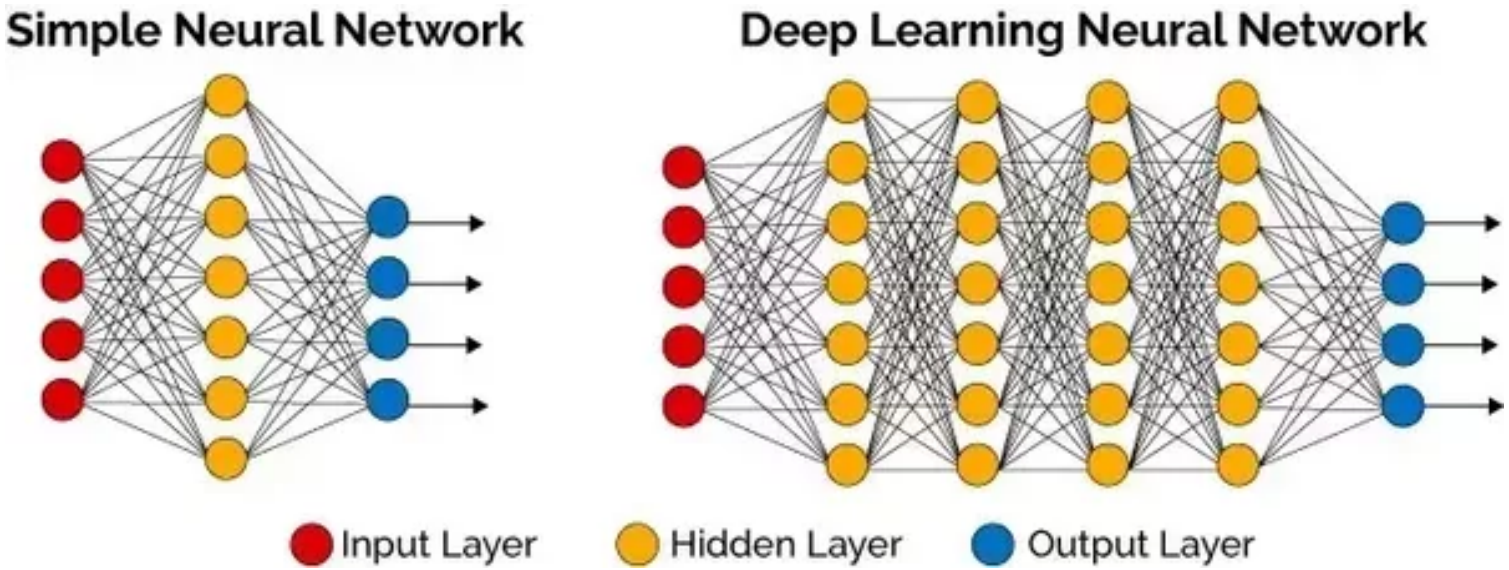
Institute of Information Science

Academia Sinica

Taiwan

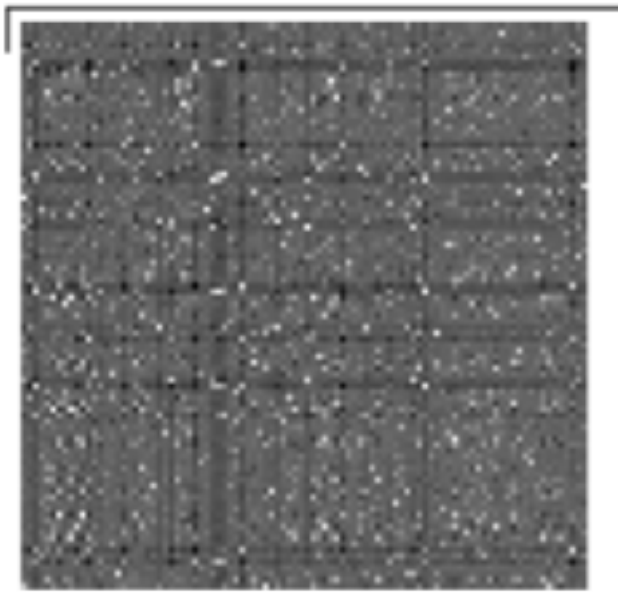
Mar 15, 2019

What do neural networks learn ?



- such that they can usually generalize
- such that transfer learning is possible
- ...

Results for between-net correlation

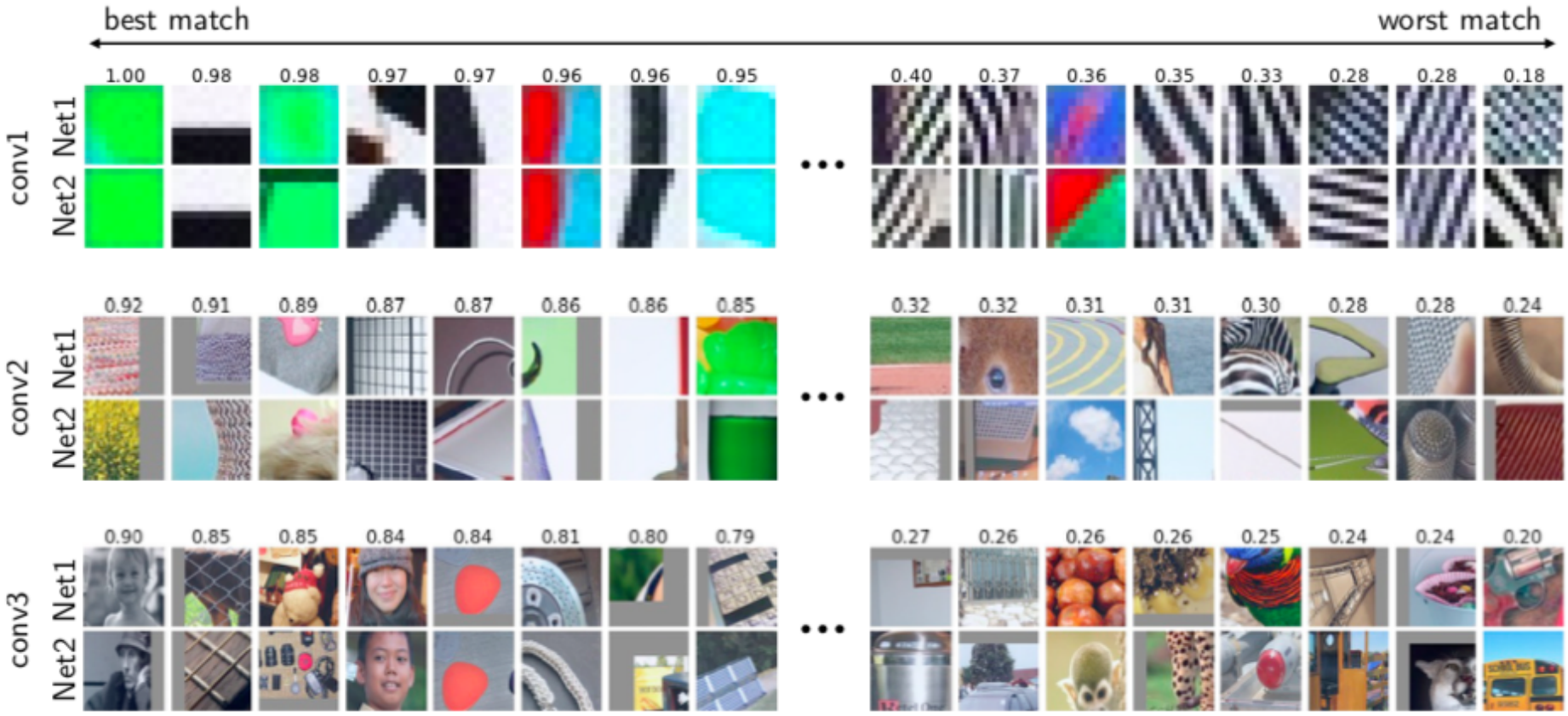


natural (unaligned) order

Is there a one-to-one alignment between features learned by different neural networks ?
If we allow ourselves to permute the units of one network, to what extent can we find equivalent or nearly-equivalent units across networks ?

bipartite semi-matching

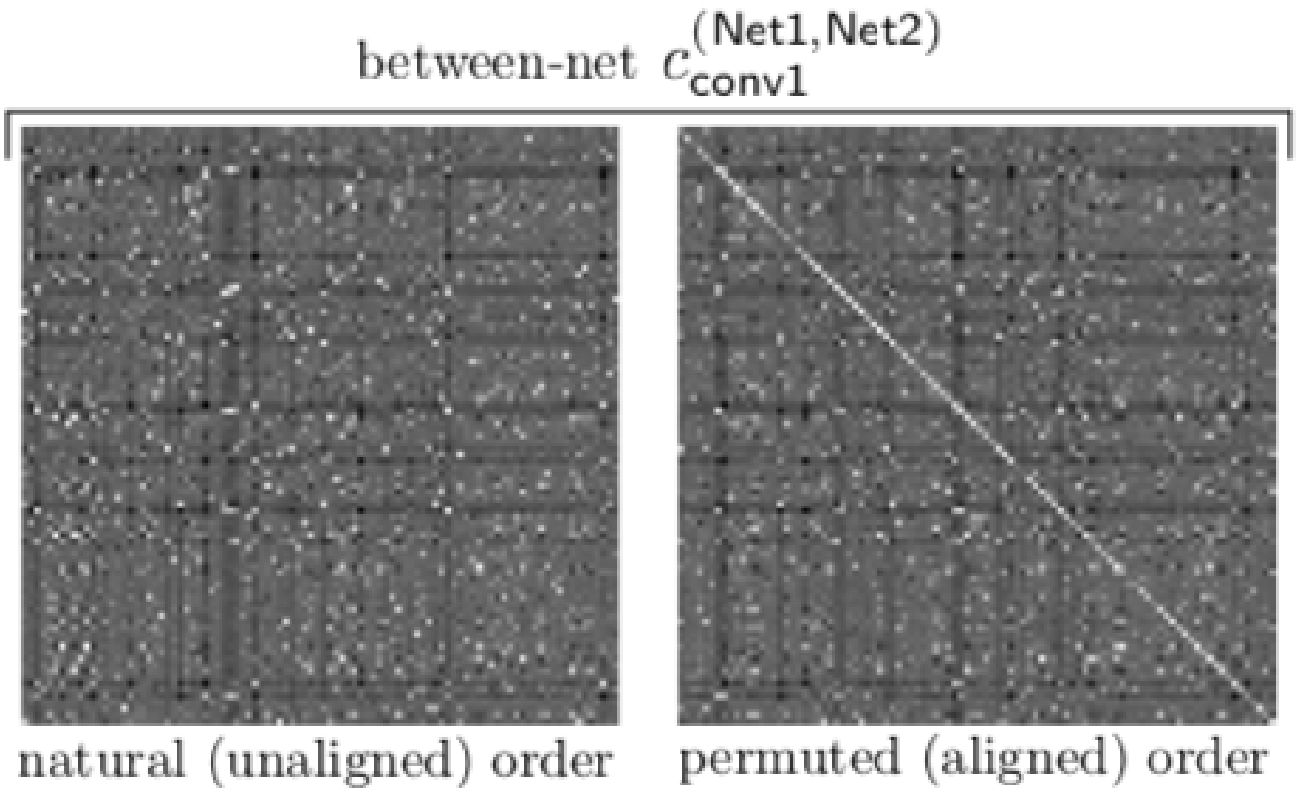
Plot the image patch from the validation set that causes the highest activation for that unit.



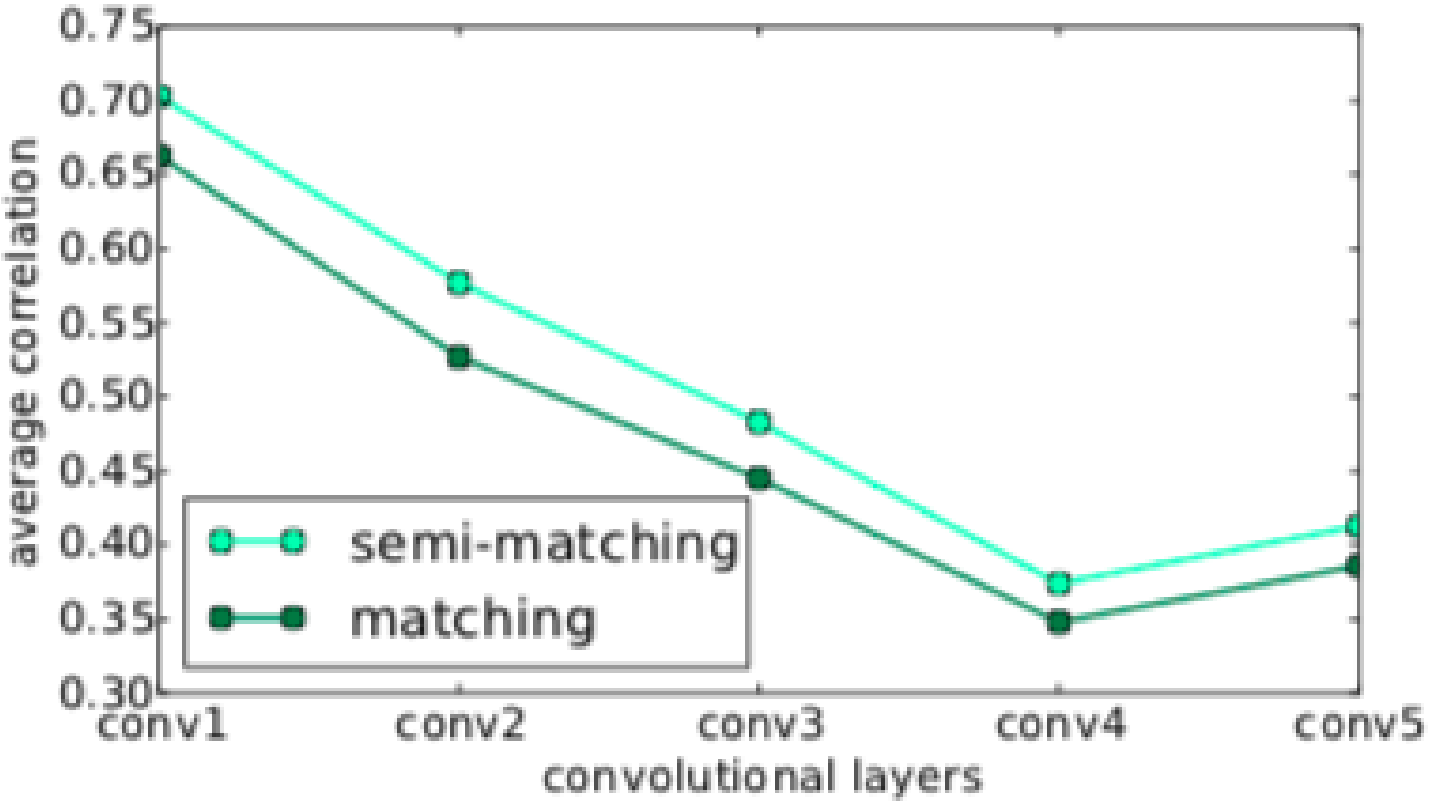
between-net correlation

bipartite matching

Find a one-to-one assignment between Net1 and Net2 such that the correlation is maximized.



between-net correlation



Average correlations between paired conv1 units in Net1 and Net2.

local v.s. distributed

How well the units of Net2 be predicted by the units of Net1 ?

- **local code** : the units that match well one-to-one
- **slightly distributed code** : the units do not match well one-to-one, but are predicted well by a sparse model
- **fully distributed** : making predictions from one network to another required a dense affine transformation

$\Rightarrow$ The representation codes are a mix between a local code and slightly, but not fully, distributed codes across multiple units.

local v.s. distributed

	Sparse Prediction Loss (after 4,500 iterations)					
	decay 0	decay 10^{-5}	decay 10^{-4}	decay 10^{-3}	decay 10^{-2}	decay 10^{-1}
conv1	**0.170**	**0.169**	**0.162**	**0.172**	0.484	0.517
conv2	**0.372**	**0.368**	**0.337**	**0.392**	0.518	0.514
conv3	0.434	0.427	0.383	0.462	0.497	0.496
conv4	0.478	0.470	0.423	0.477	0.489	0.488
conv5	0.484	0.478	0.439	0.436	0.478	0.477

Average prediction error for mapping layers with varying L1 penalties (i.e. decay terms).

SVCCA

Singular Vector Canonical Correlation Analysis (SVCCA) is a tool for quickly comparing two representations (allowing comparison between different layers and networks) in a way invariant to affine transform.

Ask

1. Is the dimensionality of a layer's learned representation the same as the number of neurons in the layer ?
2. What do deep representation learning dynamics look like ?

Method

Given dataset $X = \{x_1, \cdots, x_m\}$ and a neuron i on layer ℓ. We define $\mathbf{z}_i^\ell$ to be the vector of outputs on X :

$$\mathbf{z}_i^\ell = (\mathbf{z}_i^\ell(x_1), \cdots, \mathbf{z}_i^\ell(x_m)$$

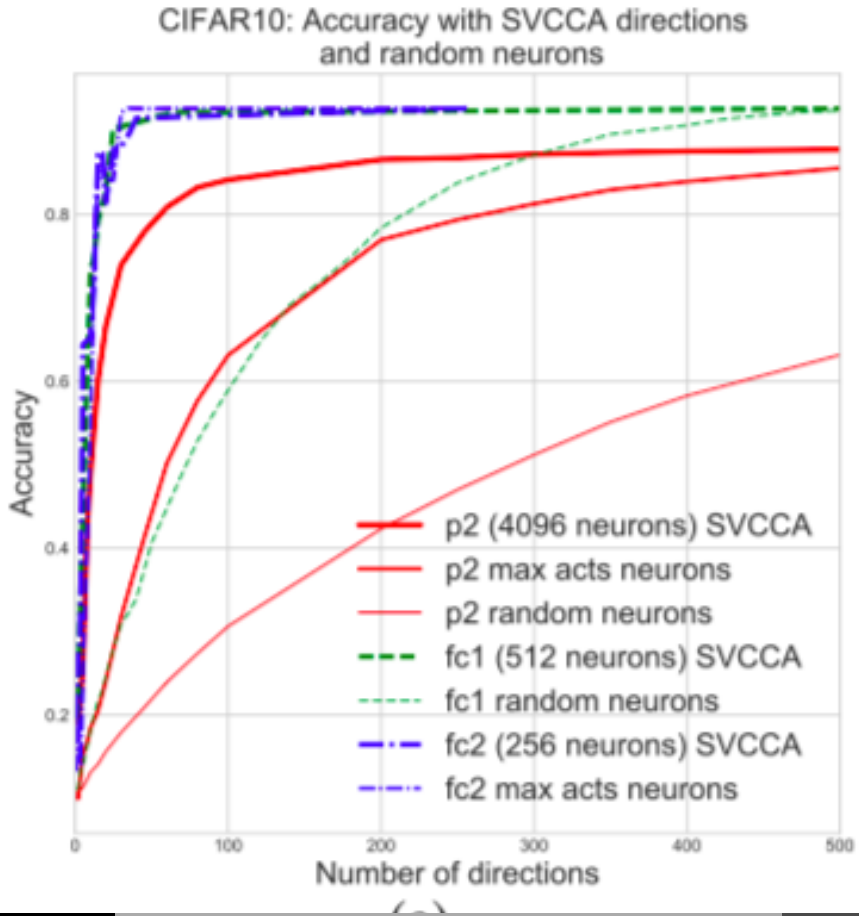
- **input** : $\ell_1 = \{\mathbf{z}_1^{\ell_1}, \cdots, \mathbf{z}_{h_1}^{\ell_1}\}$ and $\ell_2 = \{\mathbf{z}_1^{\ell_2}, \cdots, \mathbf{z}_{h_2}^{\ell_2}\}$
- **SVD** : Performs a singular value decomposition of each subspace to get subspaces $\ell_1' \subset \ell_1, \ell_2' \subset \ell_2$
- **CCA** : linearly transform $\tilde{\ell}_1 = W_1\ell_1', \tilde{\ell}_2 = W_2\ell_2'$ to be as aligned as possible and compute correlation coefficients

$$\text{corrs} = \{\rho_1, \ldots \rho_{\min(h_1', h_2')}\}$$

- **output** : $\tilde{\ell}_1, \tilde{\ell}_2$ and ρ's

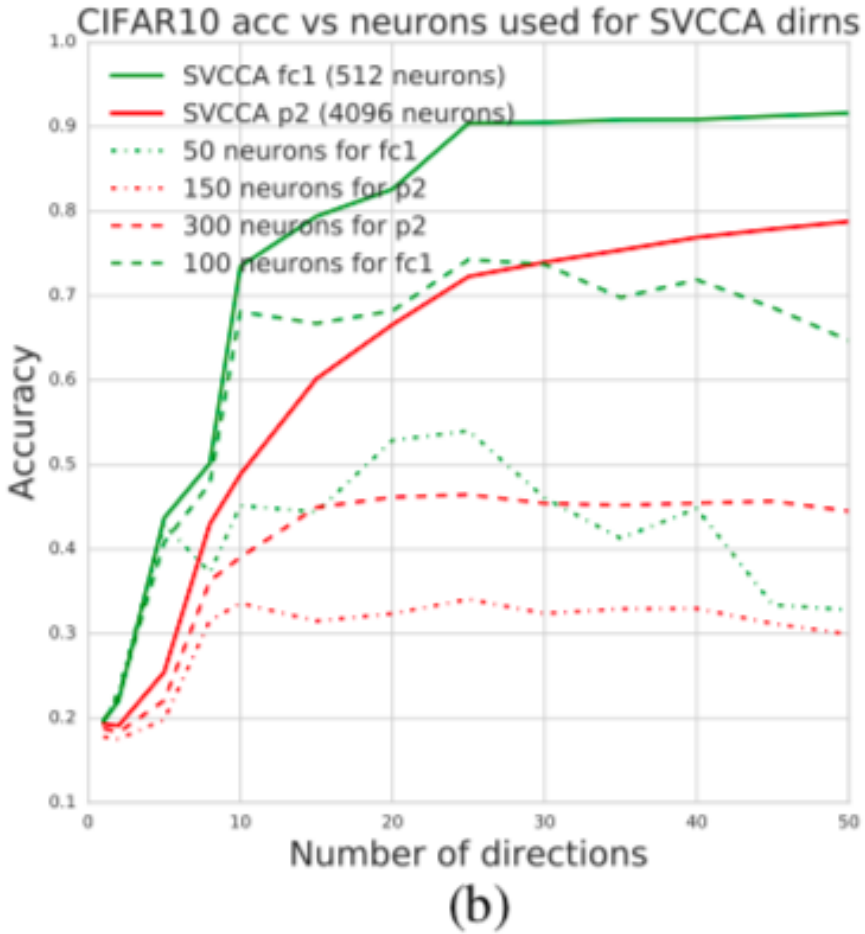
Dimensionality of learned representation

1. The subspace directions found by SVCCA are much more important to the representation learned by a layer, relative to neuron-aligned directions.



Dimensionality of learned representation

2. At least some of these directions are distributed across many neurons.



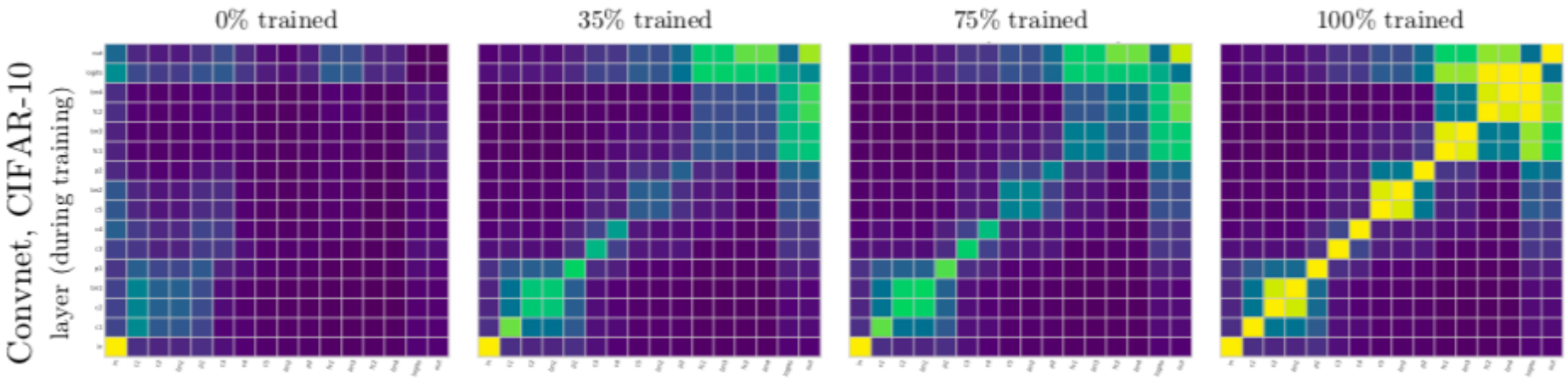
Learning dynamics

Define SVCCA similarity $\bar{\rho}$, that encapsulates how well the representations of two layers are aligned with each other,

$$\bar{\rho} = \frac{1}{min(h_1, h_2)} \sum_i \rho_i$$

- to understand learning dynamics during training process
- to understand how knowledge about the target evolves throughout the network

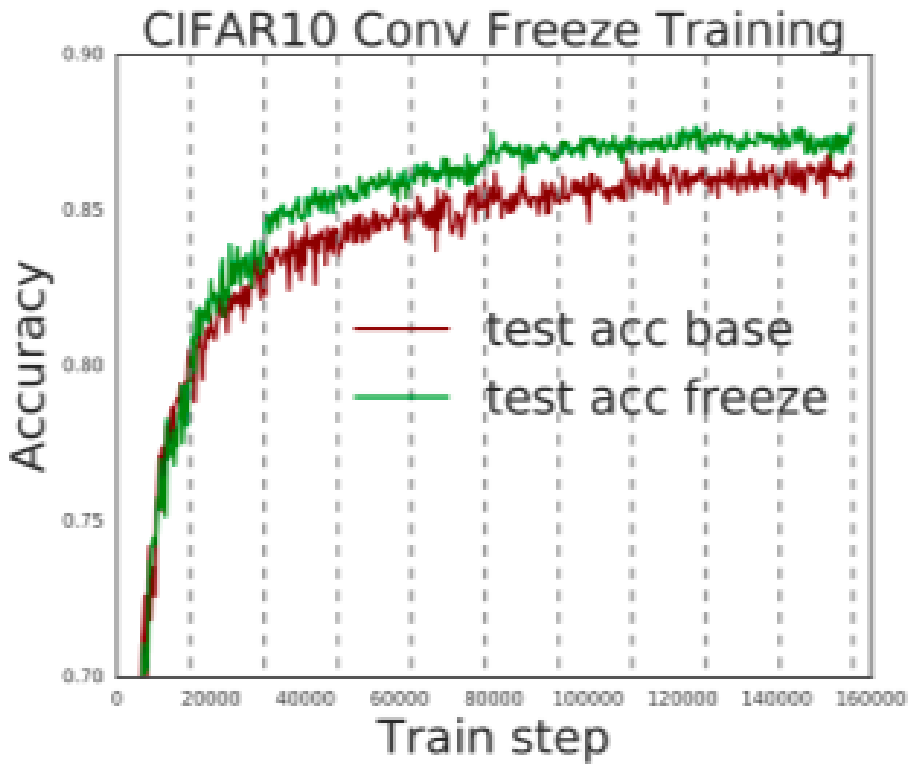
Dimensionality of learned representation

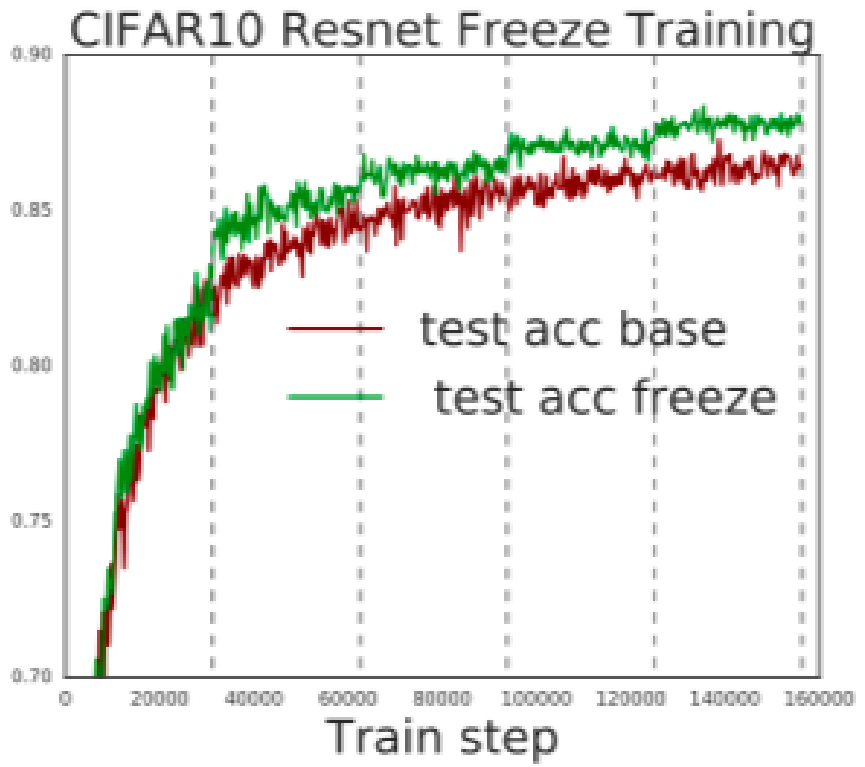


- y-axis : ℓ_1 is layer y with $0, 35, 75, 100\%$ trained.
- x-axis : ℓ_2 is layer x with 100% trained.

Dimensionality of learned representation

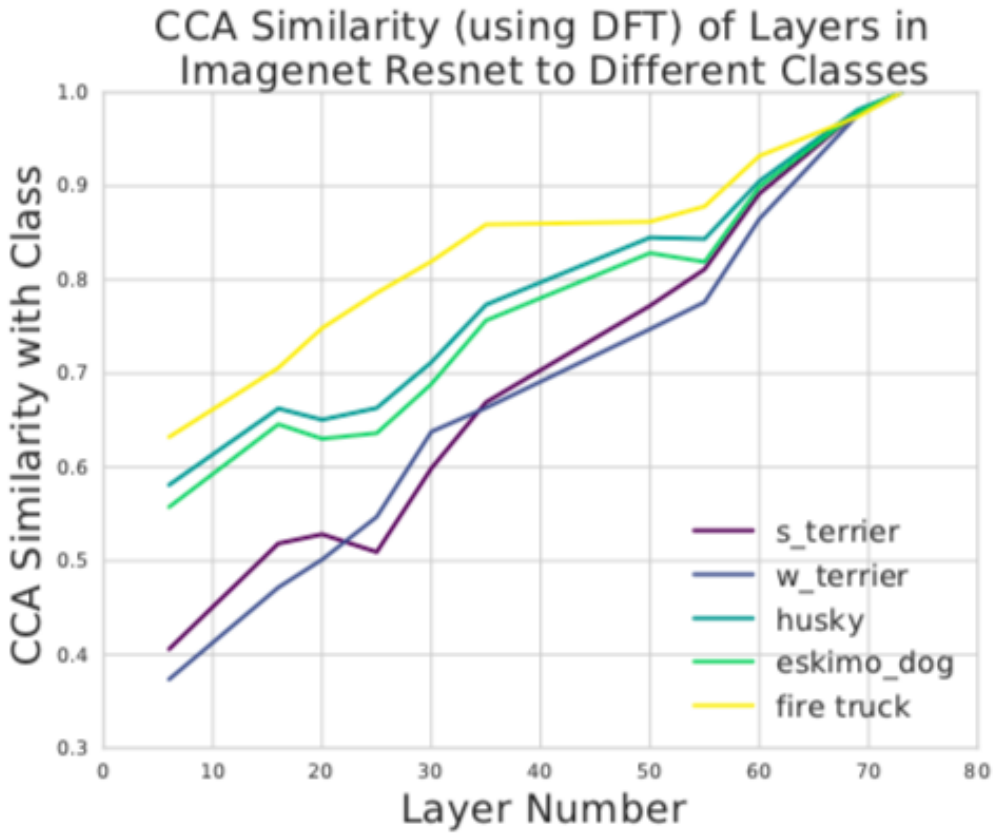
From previous figure, we know that networks broadly converge from the bottom up. Is there any benefit to **freeze** lower layers during training, only updating higher and higher layers ?





knowledge about target

Compare the CCA similarity between the logits of five classes and layers in the Imagenet Resnet. The five classes are firetruck and two pairs of dog breeds (terriers and husky like dogs).



More about representational similarity

In addition to learning dynamics, there are still much to do with SVCCA. For example

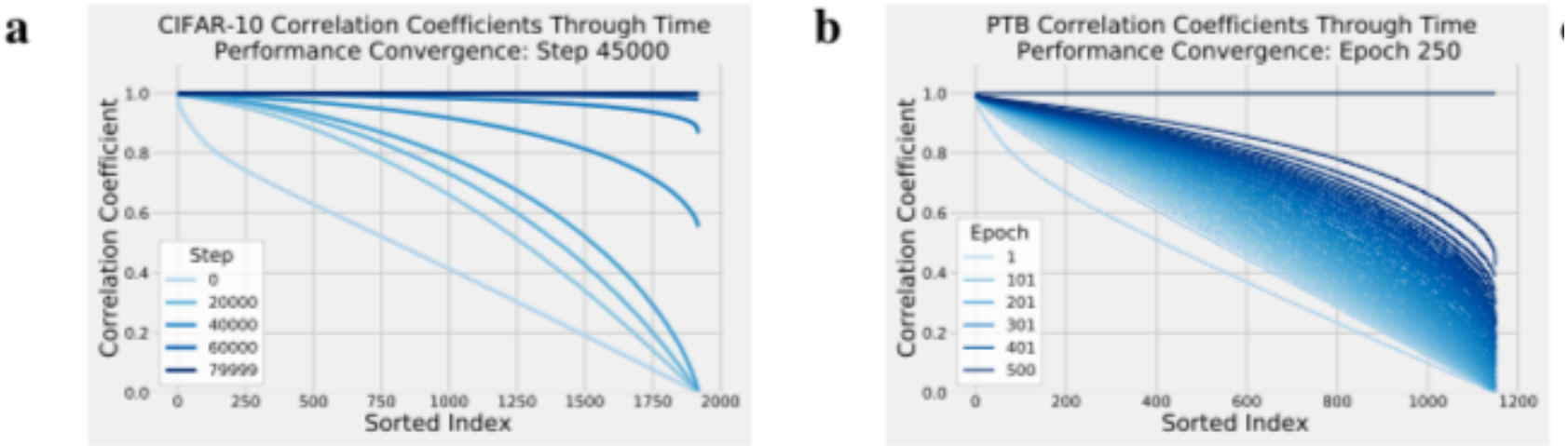
1. Do networks which generalize converge to more similar solutions than those which memorize ?
2. Do wider networks converge to more similar solutions than narrower networks ?

projection weighted CCA

Previous method measure the similarity by the **mean** correlation coefficient.

$$\bar{\rho} = \frac{1}{min(h_1, h_2)} \sum_i \rho_i$$

It implicitly assumes that all those CCA vectors are equally important to the representations. However, many of the coefficients continue to change well after the network's performance has converged.



projection weighted CCA

This result suggests that the unconverged coefficients and their corresponding vectors may represent **noise** which is unnecessary for high network performance.
The weight of vectors are chosen by the hypothesis that CCA vectors that account for a larger proportion of the original outputs are likely to be more important.
Layer ℓ_1 , have neuron activation vectors $\ell_1 = \{\mathbf{z}_1, \cdots, \mathbf{z}_{h_1}\}$. For each direction, let

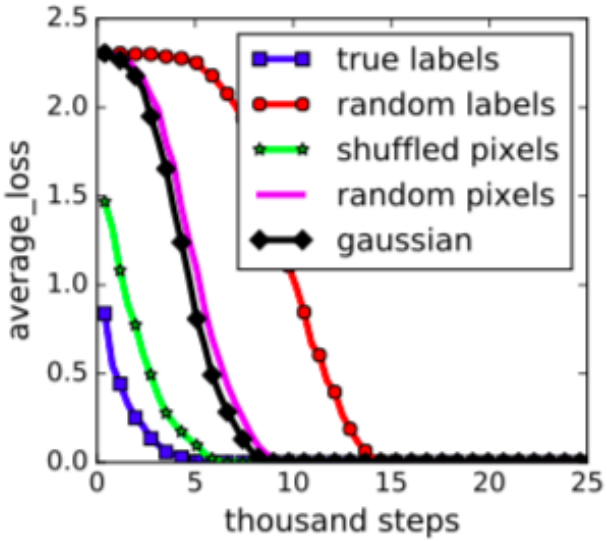
$$\tilde{\alpha}_i = \sum_j^{h_1} |\langle \tilde{\ell}_1, \mathbf{z}_j \rangle|$$

Normalize $\tilde{\alpha}$'s to get α's. Then

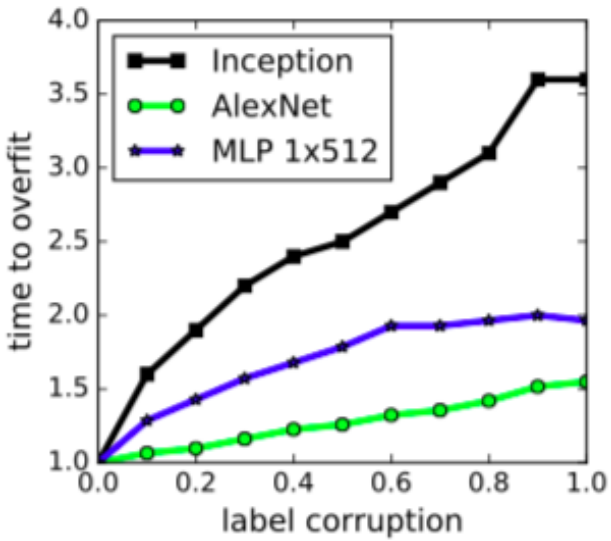
$$\bar{\rho} = \sum_i \alpha_i \rho_i$$

generalization v.s. memorization

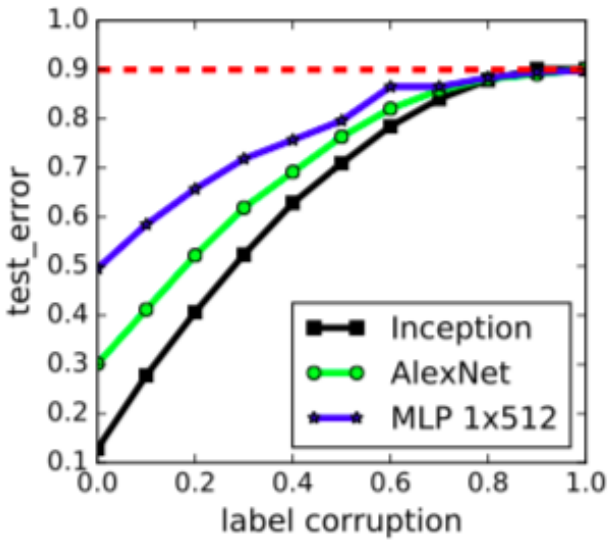
Do networks which generalize converge to more similar solutions than those which memorize ?



(a) learning curves



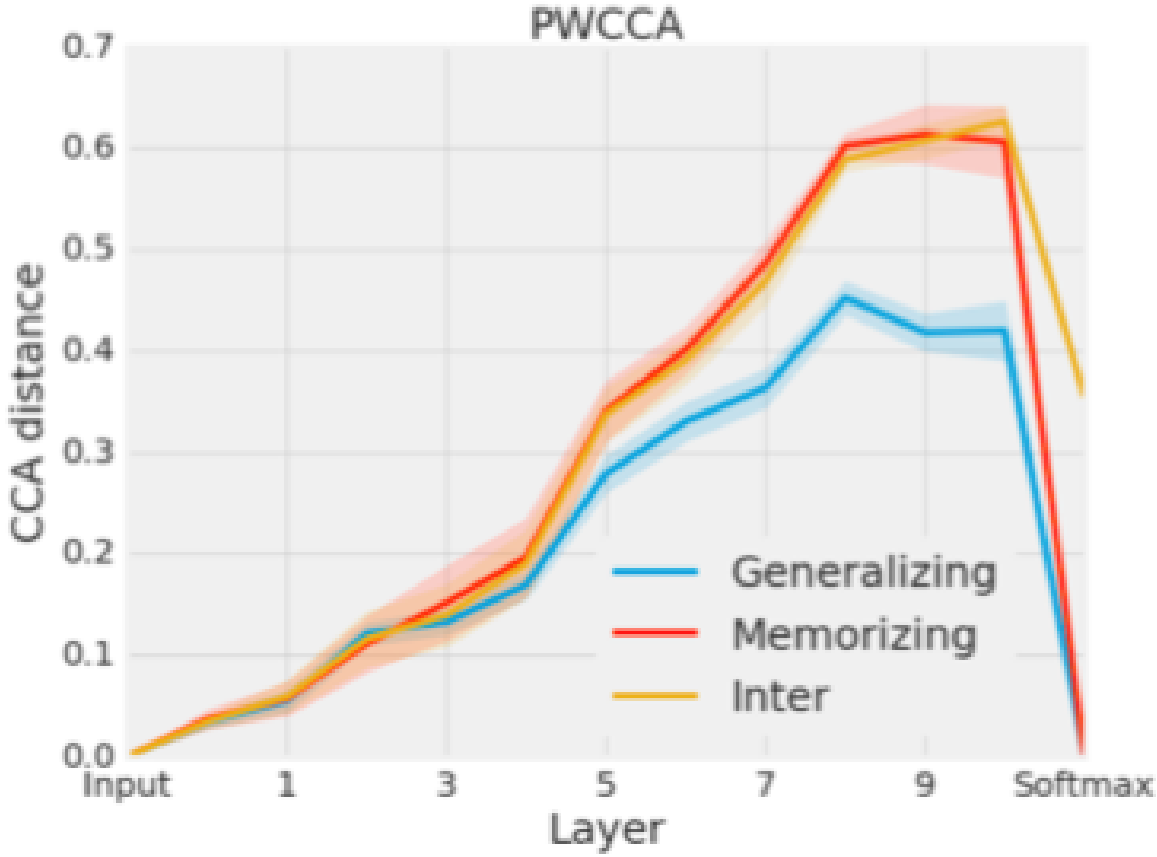
(b) convergence slowdown



(c) generalization error growth

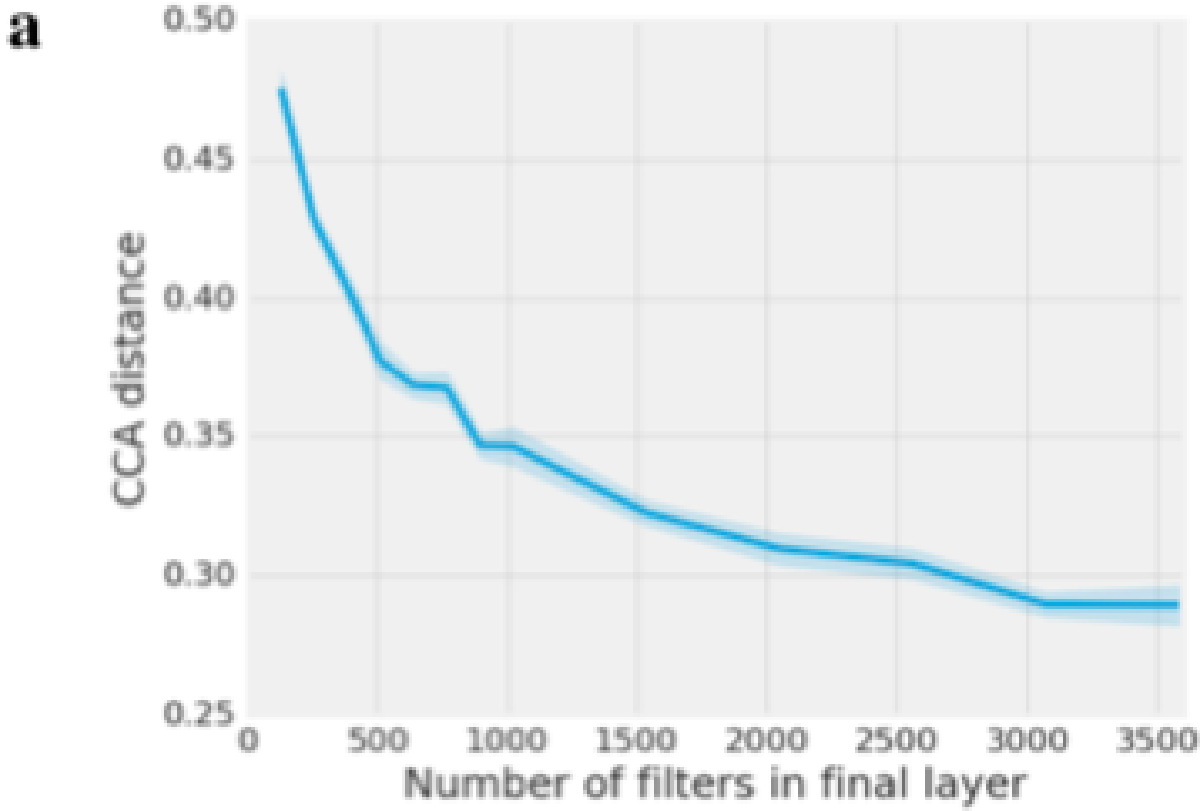
generalization v.s. memorization

Do networks which generalize converge to more similar solutions than those which memorize ?



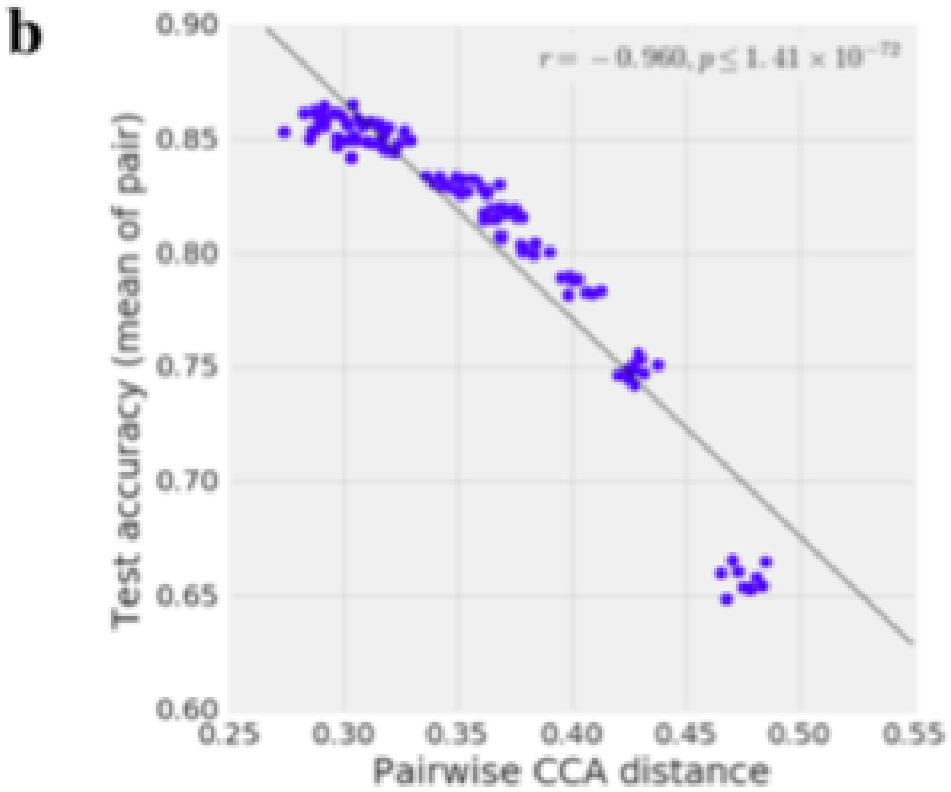
wide network v.s. narrow network

Do wider networks converge to more similar solutions than narrower networks ?



wide network v.s. narrow network

It has been observed that networks initialized and trained from the start with fewer parameters converge to poorer solutions than those derived from pruning a large networks



References

1. Convergent Learning : Do different neural networks learn the same representations ? (ICLR 2016)
2. SVCCA : Singular Vector Canonical Correlation Analysis for Deep Understanding and Improvement
3. Insights on representational similarity in neural networks with canonical correlation (NIPS 2018)
4. Understanding deep learning requires rethinking generalization (ICLR 2017)