Abstraction Mechanisms Predict Generalization in Deep Neural Networks

Alex Gain[1]
Hava Siegelmann[2]

1. The Johns Hopkins University
2. University of Massachusetts Amherst

Contributions

- We developed metrics inspired by neuroscience that are predictive of generalization and training dynamics.
- Key to our metrics is relating input complexity to deep representations.
- This provides a new direction to approaching generalization with potential tertiary applications.

Motivation

- DNNs require deeper representations in order to classify harder examples. [1]
- We formalize this as, for input x and network *net*:
  - $\alpha(x) \triangleq$ difficulty of input $\in \mathbb{R}$
  - $\beta(x, net) \triangleq$ use of deep representations $\in \mathbb{R}$
- I.e. in DNNs, for the distribution trained on:

  $\implies \alpha \propto \beta$

Motivation

- To what extent is $\alpha \propto \beta$?
  $\implies$ Let $\rho_{\alpha,\beta}$ be a measure of $\alpha \propto \beta$.
- **Hypothesis:** Higher $\rho_{\alpha,\beta}$'s correspond to "better" networks.
- **Question:** Can $\rho_{\alpha,\beta}$ predict generalization? (Assuming good definitions of α and β)

Neuroscience Inspiration

- Are there good definitions of α, β for the brain?
- In [5], neuroscience work shows that, for the brain, $\rho_{\alpha,\beta}$ is large:
  - $\alpha(T) =$ abstractness of task T, as measured by mechanical turk surveys.
  - $\beta(T) =$ use of "deeper" neurons, where "deeper" corresponds to distance from sensory cortices.

Defining α

- Back to our formalization of the result from [1]:
  - $\alpha(x) \triangleq$ difficulty of input
  - $\beta(x, net) \triangleq$ use of deep representations
- Empirically, we show $\alpha(x) \propto$ input complexity, as measured by:
  1. Compression ratio of x calculated via compression algorithms (as done in [4]).
  2. Shannon entropy estimation via histogram binning of feature values of x:

     $$-\sum_{i=1}^{N} p_{bin_i} \log p_{bin_i}$$

     for N bins where p_{bin_i} is proportional to the frequency of features in the range defined by bin i.
- Both give qualitatively similar results. We use method 2 for its ease of implementation.

Defining α

Q: Does input complexity correspond to classification difficulty?

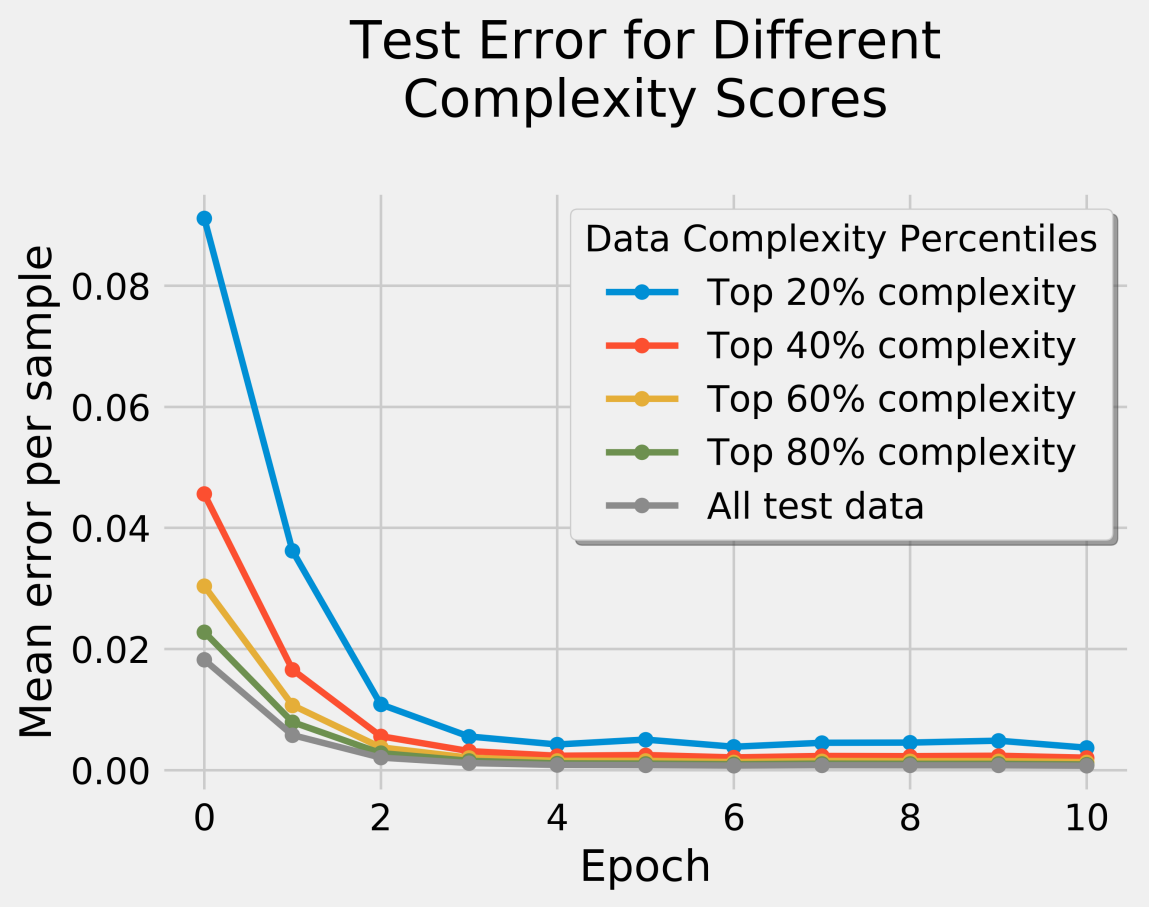

Figure: Mean test error versus training time for varying percentiles of input complexity. The model is an MLP trained on MNIST.

Defining β

- Next, we define $\beta(x)$ as:
  - The linear regressed slope on points $(d, z_d(x))$ where $d = 1, \ldots, L$ for an L layer network, where $z_d(x)$ is the sum of activation values for layer d of the network.
- β as defined will be proportional to the use of deeper neurons – it is a coarse measure of use of depth.
- Note that β does *not* model the distribution of activation values versus depth (which is highly non-linear!). It is just a coarse measure that is practical for our purposes.

Cognitive Neural Activation metric (CNA)

- The CNA (our metric for test performance) is defined for batch X and network *net*:

$$corr(\alpha(x), \beta(x, net) \mid x \in X)$$

  where *corr* is Pearson correlation (though other correlations can be used).
- Intuitively, well-performing networks should show higher CNA values.

CNA Results

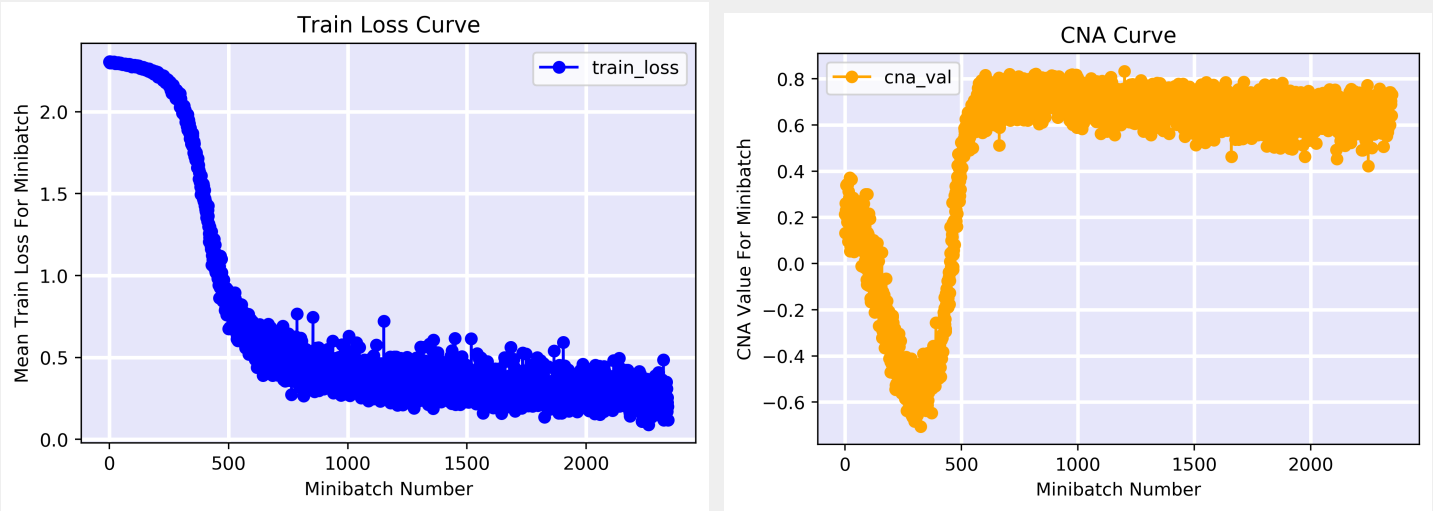

Figure: A simple MLP trained on MNIST with the training loss values (Left) and CNA values (Right) shown over training time. It is clear that the CNA shows a high correlation with training loss, with inflection points of both curves occuring at roughly the same timestep.

CNA Results

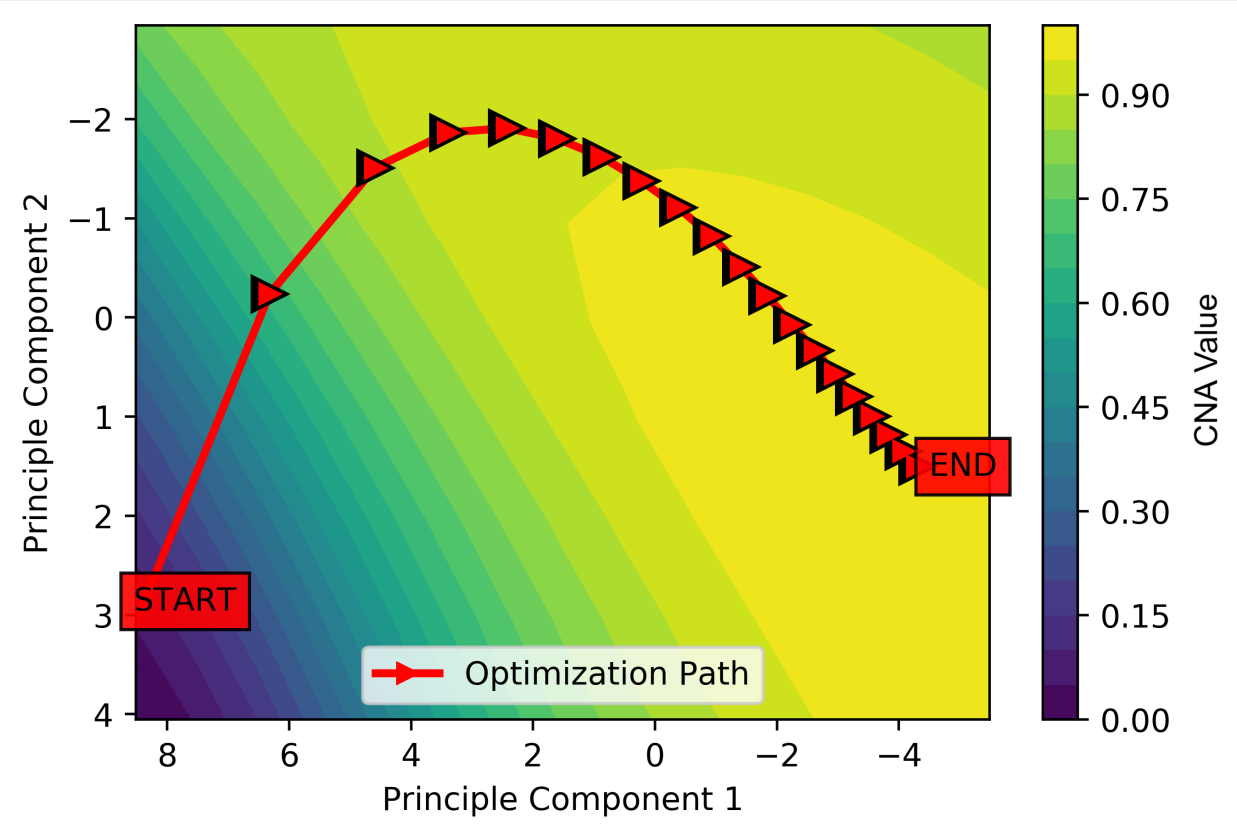

Figure: Network optimization trajectory (red curve) over training time visualized in 2D via PCA. The CNA gradient (contour background) is approximated via sampling from the principle component space and using the inverse PCA transformation.

CNA Results

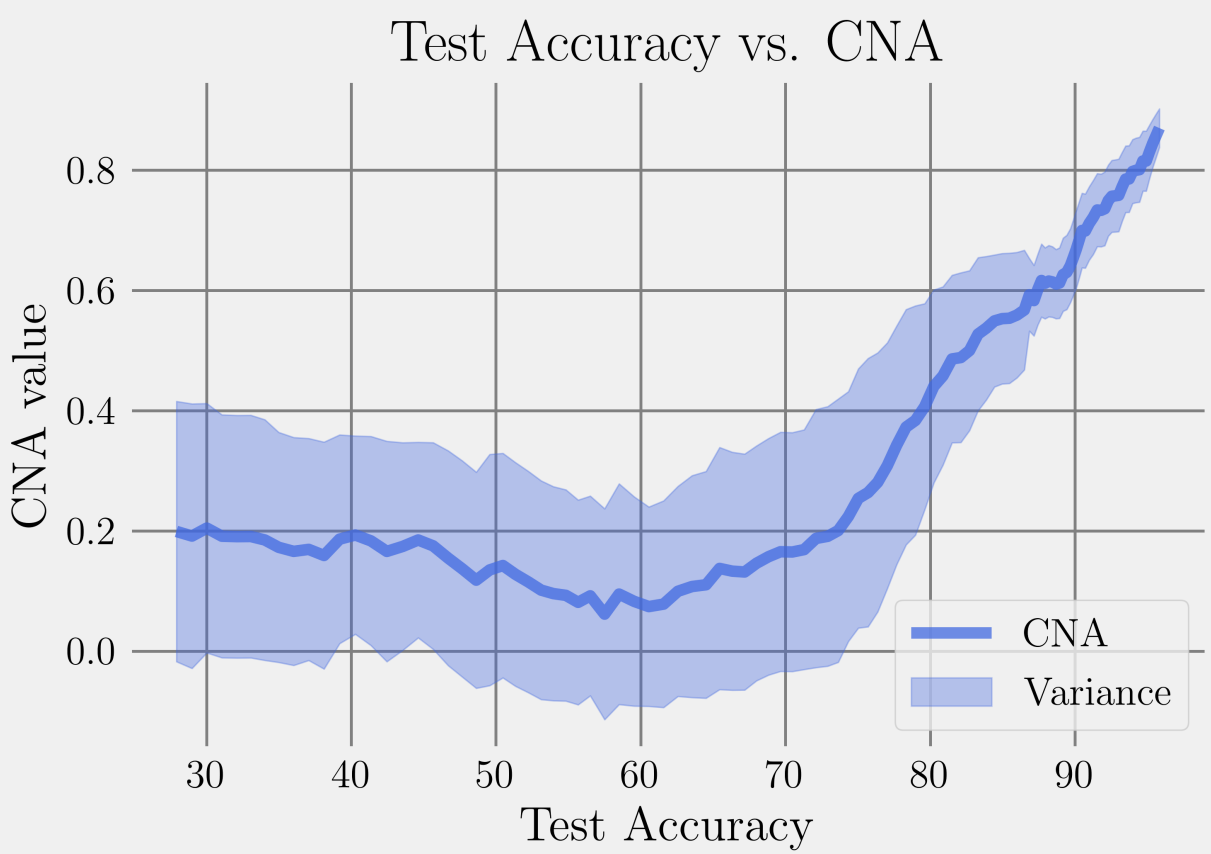

Figure: CNA vs Test Accuracy (moving average) for close to 200 network instances. Datasets: MNIST, FashionMNIST, CIFAR-10, CIFAR-100, ImageNet-32. Architectures: MLP, VGG-18, ResNet-18, ResNet-101.

CNA-Margin: Predicting the Generalization Gap

- We develop another formulation focused on measuring the generalization gap.
  - *Generalization gap: difference in performance between training and test set for a given network and data distribution.*

CNA-Margin: Predicting the Generalization Gap

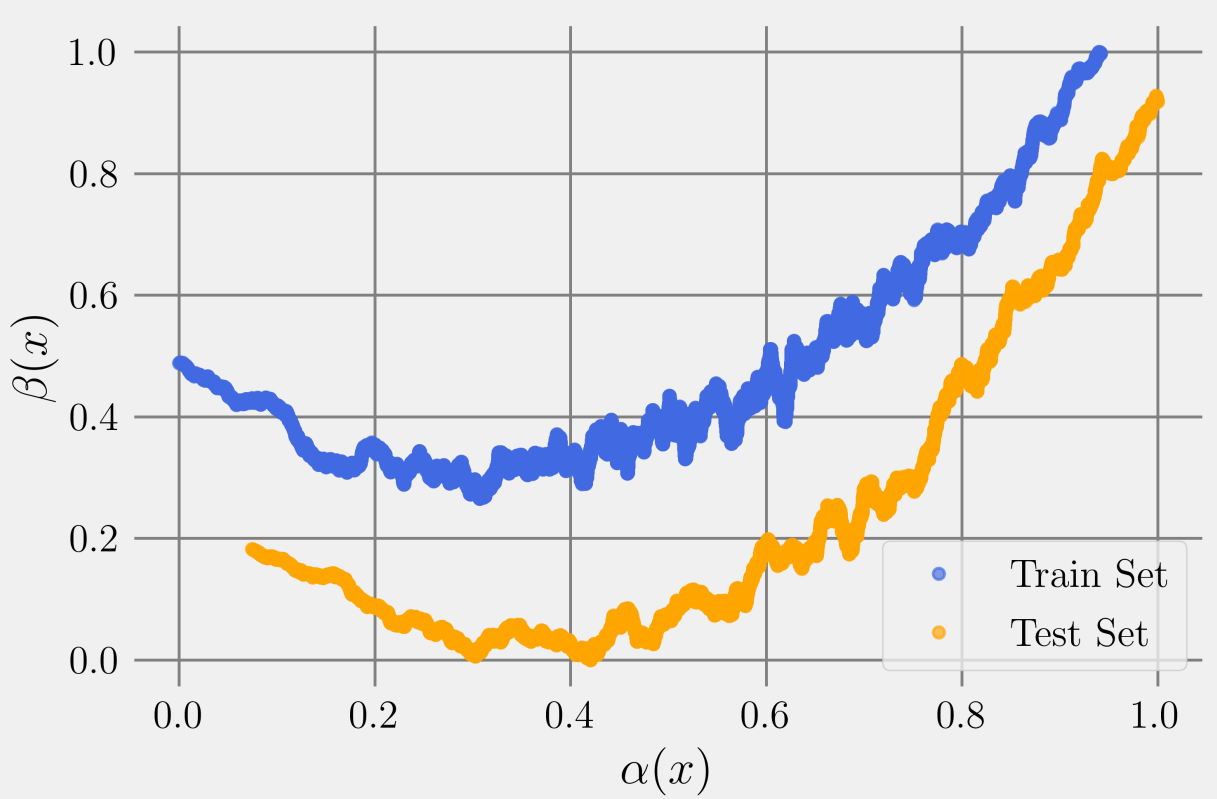

Figure: (α, β)-curves for an MLP trained on SVHN, with the train set in blue and the test set in orange. The gap between the two curves corresponds to the gap between the train and test set performance for the network.

CNA-Margin: Predicting the Generalization Gap

- We define the CNA-Area (CNA-A) as the "area" betwen the two curves shown:

$$CNA_{\mathcal{A}}(X_{\text{train}}, X_{\text{test}}) \triangleq \max_{P \in S} A(P)$$

  where $A(P)$ denotes the area of a polygon P, and S denotes the set of polygons that can be inscribed between the curves.
- The CNA-Margin (CNA-M) is simply CNA-A multiplied by the training margin γ_{margin} (see [3] for definition):

$$CNA_{\mathcal{M}}(X_{\text{train}}, X_{\text{test}}) \triangleq \gamma_{\text{margin}} \cdot CNA_{\mathcal{A}}(X_{\text{train}}, X_{\text{test}})$$

- This takes a similar form to sharpness-based generalization metrics, which are SOTA for generalization gap prediction [2], e.g. the L2 norm metric is proportional to:

$$\frac{1}{\gamma_{\text{margin}}^2} \sum_k ||W_k||_2$$

CNA-M Results

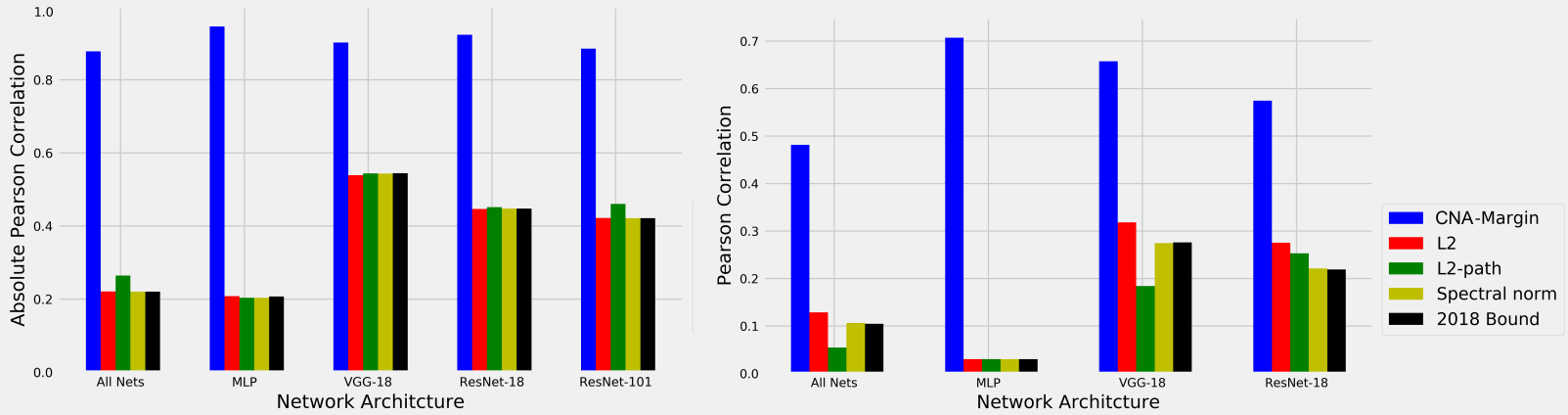

Figure: Generalization gap correlation for CNA-M and comparison generalization metrics. Left figure is w.r.t. standard datasets and a Gaussian noise dataset, right figure is w.r.t. standard datasets for varying degrees of shuffled labels (ranging from 10% to 50%).

CNA-M Results

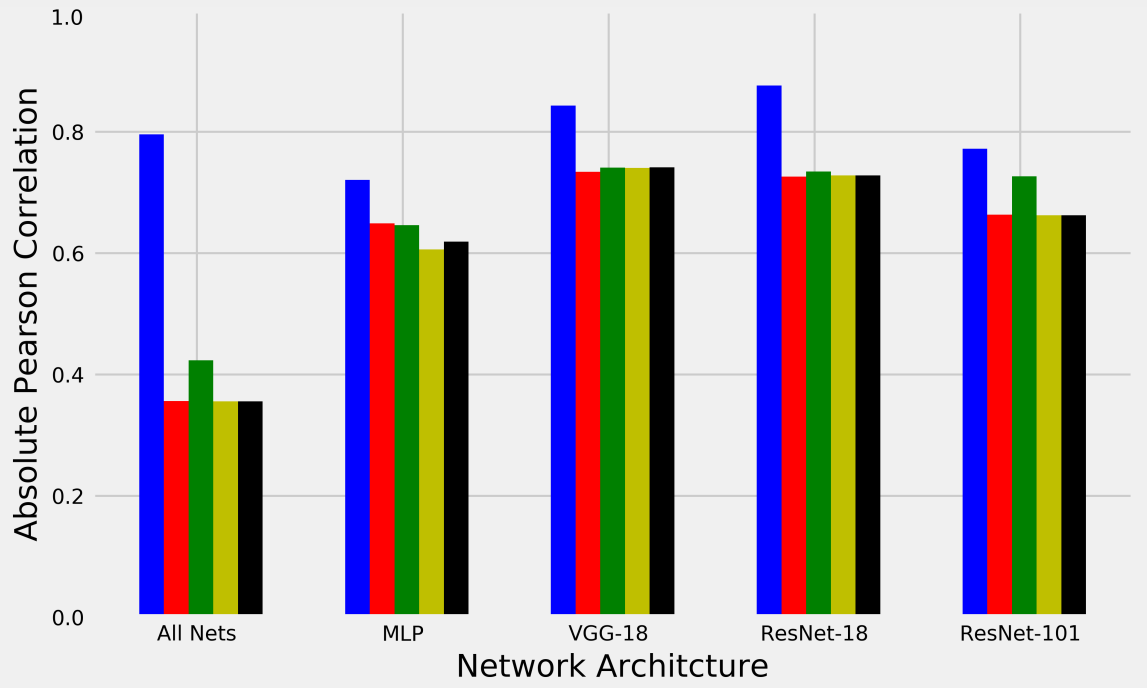

Figure: Generalization gap correlation for CNA-M and comparison generalization metrics for standard datasets and training only (i.e. no Gaussian noise dataset included). Legend in previous slide.

Summary and Conclusions

- CNA shows interesting properties w.r.t. training.
  $\implies$ Potentially useful as an unsupervised loss term and for understanding training dynamics.
- CNA-M shows excellent prediction of the generalization gap.
  $\implies$ Potentially can lead to insights into generalization not attainable by current directions.
- Tertiary contributions:
  - These nice results suggest studying DNNs under the $\alpha - \beta$ framing in general can lead to more insights or applications.
  - We have formalized and provided direct empirical measures of "deeper representations are needed in order to classify harder examples."

References

Gao Huang, Danlu Chen, Tianhong Li, Felix Wu, Laurens van der Maaten, and Kilian Q Weinberger.

Multi-scale dense networks for resource efficient image classification.
arXiv preprint arXiv:1703.09844, 2017.

Yiding Jiang, Behnam Neyshabur, Hossein Mobahi, Dilip Krishnan, and Samy Bengio.

Fantastic generalization measures and where to find them.
arXiv preprint arXiv:1912.02178, 2019.

Behnam Neyshabur, Srinadh Bhojanapalli, David McAllester, and Nati Srebro.

Exploring generalization in deep learning.
In *Advances in Neural Information Processing Systems*, pages 5947–5956, 2017.

Joan Serrà, David Álvarez, Vicenç Gómez, Olga Slizovskaia, José F Núñez, and Jordi Luque.

Input complexity and out-of-distribution detection with likelihood-based generative models.
arXiv preprint arXiv:1909.11480, 2019.

P Taylor, JN Hobbs, J Burroni, and HT Siegelmann.

The global landscape of cognition: hierarchical aggregation as an organizational principle of human cortical networks and functions.
Scientific reports, 5(1):1–18, 2015.