# On the Weak Neural Dependence Phenomenon in Deep Learning

**Jiayao Zhang**[*] **Ruoxi Jia**[†] **Bo Li**[‡] **Dawn Song**[†]

## Abstract

In this paper, we present the weak neural dependence lemma in deep neural networks (DNN) with wide fully-connected layers and arbitrary nonlinear activations. The lemma states that any two pre-activations of the same layer have arbitrarily low dependence regardless of the training stage. To the best of our knowledge, we are the first to give a formal treatment of this phenomenon aided by tools from information theory. Our simulations on MNIST/CIFAR-10 corroborate with the theory. We conclude this paper by the discussion of several implications of the lemma.

## 1 Introduction

We consider deep neural networks (DNNs) of $D$ fully connected layers (Multi-Layer Perceptrons, MLPs) with arbitrary nonlinear activation $\phi(\cdot)$. We denote by $\boldsymbol{x}_l^{(t)}$ the post-activations of layer $l$ at training time $t$, and $\boldsymbol{z}_l^{(t)}$ the pre-activations (which we colloquially refer to as neurons or neural activities). We assume weights and biases are initialized as iid zero-mean Gaussians, but the biases may be taken to be zero without loss of generality for our purpose. For ease of exposition, we assume all layers contain $N_l = N \in \mathbb{N}$ neurons except the last layer, which contains $N_D = C$ neurons and is softmax-activated, where $C$ is the number of label classes.

Previous work has utilized a similar notion of independence to model activations as iid Gaussians in untrained nets [LBN$^+$17, PLR$^+$16]. Information-theoretic methods have also been widely used in the machine learning community. Recent examples include the InfoGAN [CDH$^+$16] for training GANs in an intepretable way; L2X [CSWJ18] for model interpretation; and the not-so-recent information bottleneck method [TPB00] - albeit successes in latent-variables models [AFDM16, APF$^+$17] - whose direct extension to vanilla DNNs remains under debate [TZ15, SBD$^+$18].

In this work, we study the dependence between two neurons from the same layer, captured by the mutual information (MI) between them, and prove the weak neural dependence lemma, stating that the change of the said MI is upperbounded by the change of individual neuron entropy, a quantity we assume to be unchanged in sufficiently wide nets whence the lemma holds. Simulations of MLPs on MNIST/CIFAR-10 support the assumption and the lemma, where we observe a width larger than $100$ usually suffices; hence we expect the lemma to be useful for reasonably wide nets.

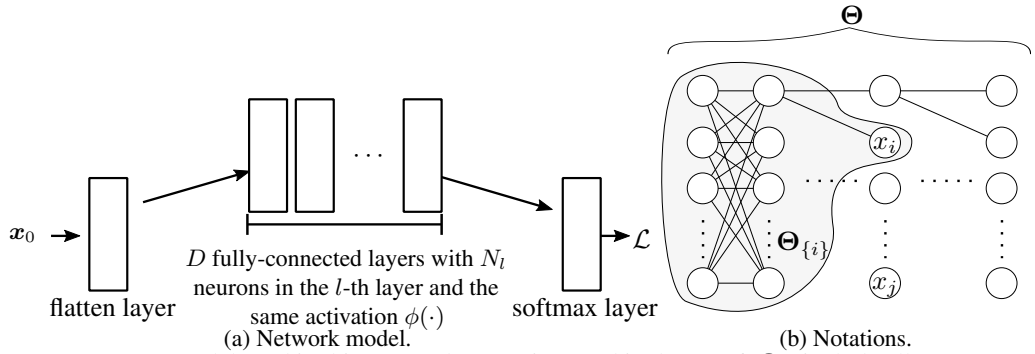
The implications of the lemma are multifold. First, the central limit theore suggests the neural activities are approximately Gaussians in the wide nets regardless of the training stage. This enables one leveraging properties of Gaussians, often much easier, to understand DNNs; second, based on the limiting behaviour of entropies, we argue that nets do not need to be too wide; lastly, although we prove the lemma for pre-activations of MLPs, the extension to post-activations can be made directly under the same proof framework.

[*]Work done while visiting UC Berkeley. University of Hong Kong, jiayaozhang@acm.org.

[†]UC Berkeley, {ruoxijia, dawnsong}@berkeley.edu.

[‡]UIUC, bli@illinois.edu.

NeurIPS '18 Workshop on Deep Learning Theory. Montréal, Canada.




(a) Network model. (b) Notations.

Figure 1: (a) MLP model used in this paper. (b) Notation used in the proof, $\boldsymbol{\Theta}_S$ include all parameters that contribute to $\{x_k\}_{k\in S}$.

## 2 Constant Entropy Assumption

The core of our result is built up on the assumption regarding the entropy of individual neurons during training. Recall the (differential) entropy of a random vector $\boldsymbol{x}$ is $H(\boldsymbol{x}) = \int p(\boldsymbol{x}) \log p(\boldsymbol{x}) \,\mathrm{d}\, \boldsymbol{x} \in \mathbb{R}$, whenever exists; and the MI between two random vectors $\boldsymbol{x}, \boldsymbol{y}$ is $I(\boldsymbol{x}; \boldsymbol{y}) = H(\boldsymbol{x}) - H(\boldsymbol{x}|\boldsymbol{y}) \geq 0$. Let $z$ be an arbitrary neuron from any layer but the last. Intuitively, if the layer containing $z$ is sufficiently large, a single neuron is not overall important, and thereby the distribution of $z$ should not differ much from its initialization. In other words, the change of $H(z)$, $\Delta H(z)$, after training, should approach to zero with infinite $N_l$:

**Assumption 1 (Constant Entropy for Neurons in Wide Layers).**

$$\lim_{N_l \to \infty} \Delta H(z) = 0. \tag{1}$$

We give the empirical validation of Assumption 1 in Section 4.

## 3 The Weak Neural Dependence Lemma

Going one step further, if a single neuron is not overall important, any two of them should not be dependent on each other much. This is formalized in the following theorem:

**Theorem 1 (Weak Neural Dependence Lemma).**

*Let $I$ be the MI of any two neurons in some layer $l$ other than the last, we have* $\lim_{N_l \to \infty} I = 0$.

For simplicity, we may sometimes write $\Delta A$ to denote the change of quantity $A$ from time 0 to $t$, and $H_i^{(t)}$ to denote $H(x_i^{(t)})$, $\boldsymbol{\Theta}$ the trainable parameters in the network, $x_i$, $x_j$ two neurons from some layer other than the last. and $\boldsymbol{\Theta}_S$ the subset of parameters from $\boldsymbol{\Theta}$ that contributes to the distribution of $x_k$ at a particular training time, for all $k \in S$. This subset is exactly all weights that in the preceding layers, union the weights in the $l$-th layer that connects to $x_k$. We illustrate this subset in Figure 1b. We write $\bar{\boldsymbol{\Theta}}_S$ to denote the complement $\boldsymbol{\Theta}/\boldsymbol{\Theta}_S$.

When $t = 0$, since $\boldsymbol{\Theta}$ is initialized as iid Gaussians, invoking the central limit theorem, neuron activities from the same layer are iid Gaussian with zero mean. Consequently, $I_{i,j}^{(0)} = 0$. The case for trained net is more involved. We first write out the MI by definition:

$$I\left(x_i^{(t)}; x_j^{(t)}\right) = H\left(x_i^{(t)}\right) - H\left(x_i^{(t)} | x_j^{(t)}\right). \tag{2}$$

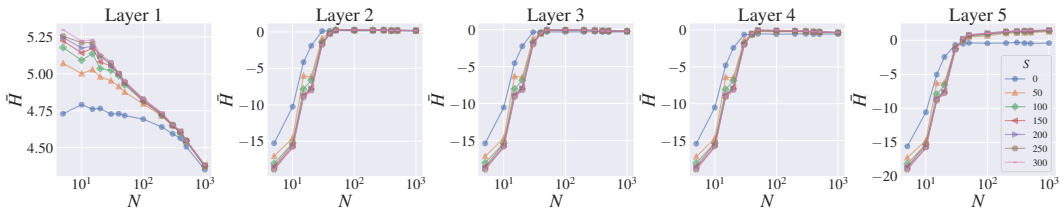
In view of the entropies, there are two sources of uncertainty, namely the data distribution, and the parameter distribution on $\boldsymbol{\Theta}$. For example, shuffling weights in a delicate manner may result a network computing the same function. Since conditioning reduces entropy, we have

$$H\left(x_i^{(t)} | x_j^{(t)}, \boldsymbol{\Theta}_{\{i,j\}}^{(t)} = \boldsymbol{\Theta}_{\{i,j\}}^{(0)}\right) \leq H\left(x_i^{(t)} | x_j^{(t)}\right). \tag{3}$$

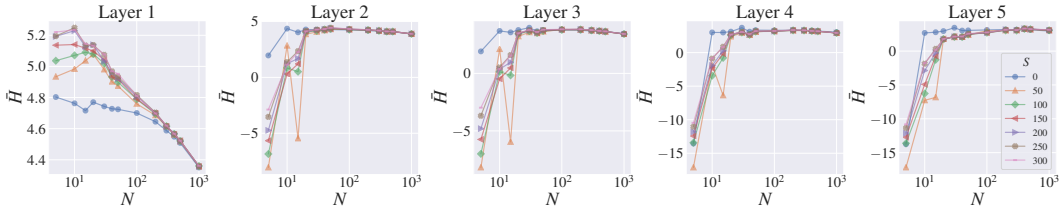
Further noted since $x_i$ is independent of all parameters that do not contribute to it during the forward pass, we have[4]

$$H\left(x_i^{(0)} | x_j^{(0)}\right) = H\left(x_i^{(t)} | x_j^{(t)}, \boldsymbol{\Theta}_{\{i,j\}}^{(t)} = \boldsymbol{\Theta}_{\{i,j\}}^{(0)}\right). \tag{4}$$

[4]See supplementary for more discussions on this step.




(a) $\phi = \tanh$.




(b) $\phi = \mathrm{ReLU}$.

Figure 2: Average neuron entropy $\bar{H}$ of each layer from a 5-layer $\phi$-activated MLP against different network width. Noted the convergence behaviour is more important whereas the actual values are data- and model-agnostic.

Combining Equations (3) and (4), we have

$$H\left(x_i^{(t)}|x_j^{(t)}\right) \geq H\left(x_i^{(0)}|x_j^{(0)}\right). \tag{5}$$

We remark that the equality $\boldsymbol{\Theta}_{\{i,j\}}^{(t)} = \boldsymbol{\Theta}_{\{i,j\}}^{(0)}$ is in distribution. For this "conditioning backward in time" to be valid, we mush show that there exists a configuration in which the network computes the same function as it is at time $t$, but with $\boldsymbol{\Theta}_{\{i,j\}}^{(t)} = \boldsymbol{\Theta}_{\{i,j\}}^{(0)}$. This is guaranteed by the universal approximation theorem [Hor91] for all layers but the last (since they have at least one trainable layers appended) such that the network is able to approximate the same function as before. On the other hand, it may be instinctive to consider if the reverse conditioning holds, that is, if we could condition on

$$H\left(x_i^{(t)}|x_j^{(t)}\right) = H\left(x_i^{(0)}|x_j^{(0)}, \boldsymbol{\Theta}_{\{i,j\}}^{(0)} = \boldsymbol{\Theta}_{\{i,j\}}^{(t)}\right). \tag{6}$$

and perform a similar argument. Unfortunately, this in general does not apply since the initial parameter distribution is considered fixed *a priori* (e.g., most commonly Gaussians) while the distribution $\boldsymbol{\Theta}_{\{i,j\}}^{(t)}$ after actually training to time $t$ may not have the same distribution.

Now combining Equations (4) and (5) yields

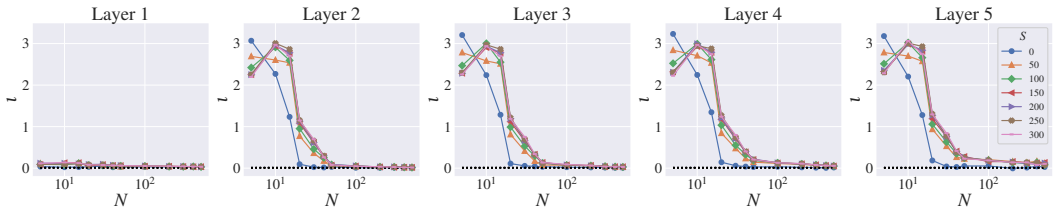
$$I\left(x_i^{(t)}; x_j^{(t)}\right) = H\left(x_i^{(t)}\right) - H\left(x_i^{(t)}|x_j^{(t)}\right) \leq H\left(x_i^{(t)}\right) - H\left(x_i^{(0)}|x_j^{(0)}\right). \tag{7}$$

Following (7), we have $\Delta I_{i;j} \leq \Delta H_i$; applying Assumption 1 yields $\lim_{N_l \to \infty} \Delta I_{i;j} \leq 0$, but $I_{i;j}^{(0)} = 0$, hence $\lim_{N_l \to \infty} I_{i;j} = 0$, completing the proof.

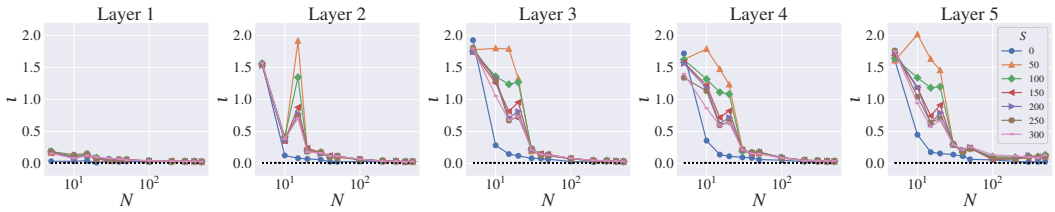
## 4 Experiments and Results

We trained a five-layer variant of the model specified in Figure 1a with either $\mathrm{ReLU}$ or $\tanh$ activation on MNIST (CIFAR-10 given in the Appendix) to convergence after 300 epochs, using stochastic gradient descent with constant learning rate $\eta = 0.001$. We selected 14 different widths ranging from 5 to 1000. For $N > 20$, The $\mathrm{ReLU}$ activated nets achieve $> 99\%$ training accuracy and $> 95\%$ validation accuracy; the $\tanh$-activated nets $> 95\%$ and $> 92\%$ respectively. However, with $5 \leq N < 20$, the performances were usually worse. We used a fixed random sample of 1000 training images (100 per class) to estimate the entropies and the MI, where the details are left to the Appendix.

We present in Figure 2 the average neuron entropy $\bar{H}$ of different layers for $\tanh$ (Figure 2a) and $\mathrm{ReLU}$ (Figure 2b) activated nets. Noted with sufficiently large $N$, $\Delta H$ becomes insignificant,




(a) $\phi = \tanh$.




(b) $\phi = \mathrm{ReLU}$.

Figure 3: Dependence score $\iota$ of each layer for different activations against different network width.

supporting Assumption 1. To obtain an overall picture of the MI, we define the dependence score $\iota_l$ to be the average pairwise MI in the $l$-th layer. We show in Figure 3 the score $\iota$ against the network width. Given two independent standard Gaussian sample with $1000$ points, the MI estimator gives $0.012 \pm 0.042$ over $1000$ trials, hence we may roughly view $0.01$ as a threshold for independence (depicted as the black dotted line in Figure 3). Noted the MIs of all layers but the last converged to the vicinity of this threshold, indicating neurons are nearly independent. Intuitively, since the last layer is unnormalized probabilities, there *should* be dependence between them: higher likelihood in one class results lower in another. More concretely in this example, a width larger than $200$ usually suffices for the theory, and we expect in practice the lemma works well for reasonably wide nets.

## 5 Consequences of the Lemma

We offer a short discussion on the consequences of the lemma in this section.

**Applicable to pre-activations.** The extension of the lemma to pre-activations can be made with ease by redoing the proof. In $\tanh$-activated nets, the extension is even simpler since MI is invariant to reparameterization given by homeomorphisms (smooth invertible maps), in particular $\tanh^{-1}$.

**Neural activities as multi-variate Gaussians.** When one looks at a particular a network at some training stage, by central limit theorem, the activations are approximately Gaussians. This approximation have been used in multiple previous work [PLR$^+$16, LBN$^+$17] in untrained nets, our results further suggest that the same holds for trained MLPs given reasonably wide layers.

**Layers do not have to be too wide.** The average individual neuron entropy exhibited a convergence behaviour, and is correlated with the training accuracy (not shown). Information-theoretically speaking, the average entropy characterizes the uncertainty in neural activities, and may relate to the capacity in terms of the the uncertainty/diversity of hidden features. Increasing layer width after convergence at a high cost of computation, therefore, may not induce further gain.

## 6 Conclusions

In this paper, we presented and proved the weak neural dependence lemma for wide MLPs. We tested this lemma on MNIST/CIFAR-10 and discussed several implications, which may of independent interest. Future directions may include the study of the underlying mechanism that more rigorously justify and prove Assumption 1 and theorem 1, strengthening the lemma from pairwise independence to general independence, as well more exhaustive experiments on different architectures.

## References


[AFDM16] Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. Deep variational information bottleneck. *arXiv preprint arXiv:1612.00410*, 2016.

[APF+17] Alexander A Alemi, Ben Poole, Ian Fischer, Joshua V Dillon, Rif A Saurous, and Kevin Murphy. An information-theoretic analysis of deep latent-variable models. *arXiv preprint arXiv:1711.00464*, 2017.

[CDH+16] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems*, pages 2172–2180, 2016.

[CSWJ18] Jianbo Chen, Le Song, Martin J Wainwright, and Michael I Jordan. Learning to explain: An information-theoretic perspective on model interpretation. In *International Conference on Machine Learning (ICML)*, 2018.

[Hor91] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2):251–257, 1991.

[KSG04] Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. Estimating mutual information. *Physical Review E*, 69(6):066138, 2004.

[LBN+17] Jaehoon Lee, Yasaman Bahri, Roman Novak, Samuel S Schoenholz, Jeffrey Pennington, and Jascha Sohl-Dickstein. Deep neural networks as gaussian processes. *arXiv preprint arXiv:1711.00165*, 2017.

[Pan03] Liam Paninski. Estimation of entropy and mutual information. *Neural computation*, 15(6):1191–1253, 2003.

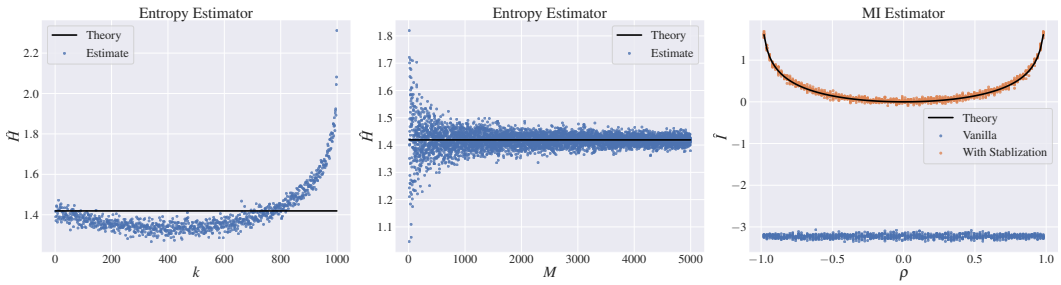
[PLR+16] Ben Poole, Subhaneil Lahiri, Maithra Raghu, Jascha Sohl-Dickstein, and Surya Ganguli. Exponential expressivity in deep neural networks through transient chaos. In *Advances in neural information processing systems*, pages 3360–3368, 2016.

[SBD+18] Andrew Michael Saxe, Yamini Bansal, Joel Dapello, Madhu Advani, Artemy Kolchinsky, Brendan Daniel Tracey, and David Daniel Cox. On the information bottleneck theory of deep learning. In *International Conference on Learning Representations (ICLR)*, 2018.

[SDBR15] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. In *International Conference on Learning Representations (ICLR) Workshop*, 2015.

[TPB00] Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

[TZ15] Naftali Tishby and Noga Zaslavsky. Deep learning and the information bottleneck principle. In *Information Theory Workshop (ITW), 2015 IEEE*, pages 1–5. IEEE, 2015.




(a) Entropy estimation varying $k$. (b) Entropy estimation varying $M$. (c) MI Estimation example.

Figure A.1: Examples of estimating entropy and MI. (a) Entropy estimate of a standard Gaussian sample of size 1000, varying the parameter $k$; (b) Fixing $k = 3$, varying the sample size $M$ of standard Gaussian samples; (c) Consider a bivariate Guassian, where the variance of one variable is $10^8$ higher than the other, stabilization by the sample standard deviation reduces the bias in estimation.

## Appendix

Note: `Python` code for reproducing all results in this paper will be released upon publication.

## A Estimating Differential Entropy and Mutual Information

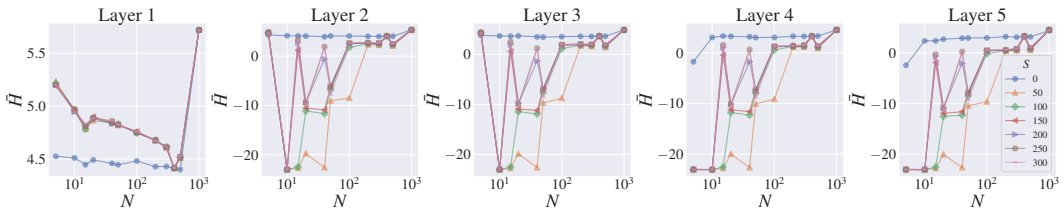
Estimating information-theoretic quantities in the case of continuous random variables has attracted many attention in research and is general perceived to be notoriously hard in high dimensions. In this paper, we stick to a simple $k$-NN based estimator that have been proven consistent and is usually good enough in practice for moderately large sample size [KSG04]. For more discussion on this topic, we recommend inquisitive readers to consult [Pan03] and subsequent work.

We now discuss in details the estimating procedure. For a set of data points $\{\boldsymbol{x}_i\}_{i=1}^M$ in $\mathbb{R}^d$ equipped with $L_\infty$ norm, the MI estimator gives

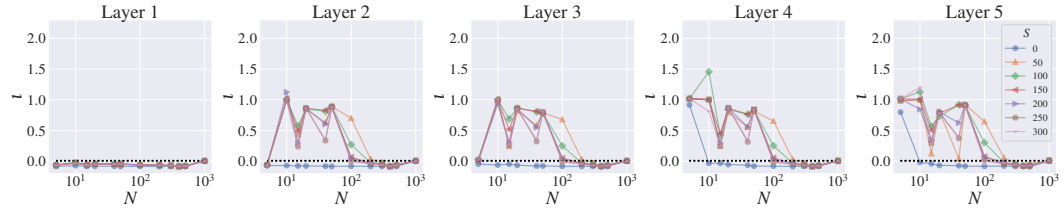
$$\hat{H}(\boldsymbol{x}) = \psi(M) - \psi(k) + \frac{d}{M}\sum_{i=1}^{M} \log \epsilon_i, \tag{A.1}$$

where $\psi(\cdot)$ is the digamma function, and $\epsilon_i$ is twice the distance between $\boldsymbol{x}_i$ and its $k$-nearest neighbors. In practice, we follow the guidelines in [KSG04] to set $2 \leq k \leq 5$. We estimate the MI by the "$3H$" method, i.e., $I(\boldsymbol{x}; \boldsymbol{y}) = H(\boldsymbol{x}) + H(\boldsymbol{y}) - H(\boldsymbol{x}, \boldsymbol{y})$ [KSG04].

We found in our experiments that the choice of $k$ is in general less important so long as we follow the recommendations; and a sample size of around 1000 strikes the balance between estimation bias and computation burden. As an illustrative example, we consider a random standard Gaussian sample of size $M$, where we present in Figure A.1a the effects of $k$ (with $M = 1000$), and in Figure A.1b the sample size $M$ (with $k = 3$). Noted the biases are tolerable with $M = 1000$ and $k = 3$. Nonetheless, we must remark on a few nuances here regarding estimating MI. In general, if $\operatorname{Var} x$ and $\operatorname{Var} y$ do not differ much, we found the estimator $\hat{I}$ gives a faithful estimate in general (not shown here). However, in the rare case when $\operatorname{Var} x \lll \operatorname{Var} y$, $\hat{I}$ may yield a negative estimate. To reduce the bias in this case and make the estimate legitimate, we stabilize each set of samples by its standard deviation. In theory, this should not alter the value of the MI, which, as we remarked in Section 5, is invariant to homeomorphisms, in particular scaling by a scalar. Indeed, consider the example of a sample of size 1000 from some bivariate Guassian $(x, y)$, where $\sigma_x = 0.0001$ and $\sigma_y = 1$. We illustrate in Figure A.1c the estimates from the vanilla estimator and the one with stabilization, where we vary the correlation coefficient $\rho$ between $x$ and $y$. Noted the vanilla estimator severely underestimated the MI; with stabilization, the estimator is able to provide a more accurate estimate of the theoretical MI (given in black line). Although this pathological example may be seemingly rare in practice, such wildly-behaved neuron pairs may indeed emerge when data become more diverse (e.g., CIFAR-10).




(a) Average entropy $\bar{H}$.




(b) Dependence score $\iota$.

Figure B.1: Simulation results on CIFAR-10. We used the same model architecture (ReLU activation) and training configurations as in the MNIST. The observations we made previously in Section 4 still apply.

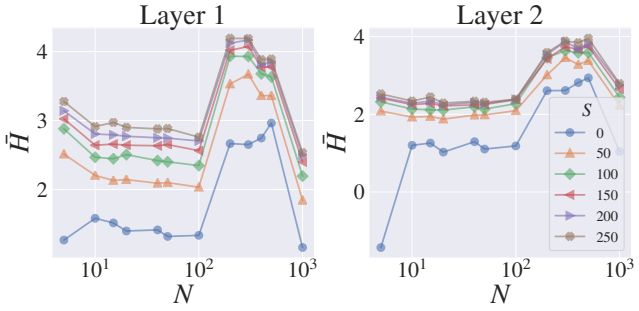



Figure B.2: Average entropy $\bar{H}$ from two dense layers of a convolutional net trained on CIFAR-10.

## B Simulation Results of MLP and Convolutional Nets on CIFAR-10

In this section, we demonstrate the simulation on CIFAR-10. We used the same model architecture as we did on MNIST with ReLU activation, and the same training configurations. We present the results in Figure B.1. Noted the average entropy exhibits a similar convergence behaviour (Figure B.1a) required by Assumption 1 thus suggesting Theorem 1 should be applicable to this case. Indeed, in Figure B.1b, we observe a similar weak neural dependence phenomenon, and noted in low width regimes, the behaviour of the dependence score $\iota$ is less regular. We remark that even after stabilization, on some occasion, the estimator $\hat{I}$ gave a value slightly less than zero. This may be in part due to the insufficiently large sample size as CIFAR-10 is more diverse. Nonetheless, this does not affect the conclusion since those values outputted by $\hat{I}$ are still within a tolerable margin at the vicinity of 0.

We also did a pilot simulation on convolutional nets trained on CIFAR-10. We used a variant of the "all-convolutional-nets" [SDBR15], which two ReLU-activated dense layers are appended after five convolutional layers. We tested Assumption 1 on those two dense layers, and found the assumption seems not apply. Hence future studies may focus on the applicability of the assumptions and the lemma to the convolutional nets.

## C Discussion on the Proof of Theorem 1.

In the proof of Theorem 1, we have stated the following in (4):

$$H\left(x_i^{(0)}|x_j^{(0)}\right) = H\left(x_i^{(t)}|x_j^{(t)}, \boldsymbol{\Theta}_{\{i,j\}}^{(t)} = \boldsymbol{\Theta}_{\{i,j\}}^{(0)}\right). \tag{C.1}$$

For this equation to make sense, we have used that

$$H\left(x_i^{(0)}|x_j^{(0)}, \mathbf{\Theta}_{\{i,j\}}^{(t)} = \mathbf{\Theta}_{\{i,j\}}^{(0)}\right) = H\left(x_i^{(t)}|x_j^{(t)}, \mathbf{\Theta}_{\{i,j\}}^{(t)} = \mathbf{\Theta}_{\{i,j\}}^{(0)}\right), \tag{C.2}$$

and one may drop the conditioning at initialization since we have prescribed the parameter initialization scheme, we thus have (C.1).