A TENSOR ANALYSIS ON DENSE CONNECTIVITY VIA CONVOLUTIONAL ARITHMETIC CIRCUITS

Anonymous authors
Paper under double-blind review

ABSTRACT

Several state of the art convolutional networks rely on inter-connecting different layers to ease the flow of information and gradient between their input and output layers. These techniques have enabled practitioners to successfully train deep convolutional networks with hundreds of layers. Particularly, a novel way of interconnecting layers was introduced as the Dense Convolutional Network (DenseNet) and has achieved state of the art performance on relevant image recognition tasks. Despite their notable empirical success, their theoretical understanding is still limited. In this work, we address this problem by analyzing the effect of layer interconnection on the overall expressive power of a convolutional network. In particular, the connections used in DenseNet are compared with other types of inter-layer connectivity. We carry out a tensor analysis on the expressive power inter-connections on convolutional arithmetic circuits (ConvACs) and relate our results to standard convolutional networks. The analysis leads to performance bounds and practical guidelines for design of ConvACs. The generalization of these results are discussed for other kinds of convolutional networks via generalized tensor decompositions.

1 INTRODUCTION

Recently, densely connected networks such as FractalNet (Larsson et al., 2016), ResNet (He et al., 2016), and DenseNet (Huang et al., 2016), have obtained state of the art performance on large problems where highly deep network configurations are used. Adding dense connections between different layers of a network virtually shortens its depth, thus allowing a better flow of information and gradient through the network. This makes possible the training of highly deep models. Models with these types of connections have been successfully trained with hundreds of layers. More specifically, DenseNets have achieved state of the art performance on the CIFAR-10, CIFAR-100, SVHN, and ImageNet datasets, using models of up to 1 thousand layers in depth. Nevertheless, whether these connections provide a fundamental enhancement on the expressive power of a network, or just improve the training of the model, is still an open question. In Huang et al. (2016), DenseNet models with 3 times less parameters than its counterpart (ResNets) were able to achieve the same performance on the ImageNet challenge. Moreover, a theoretical understanding of why the connections used by DenseNets lead to better performance compared with FractalNets or ResNets is still pending.

Despite the popularity of these models, there are few theoretical frameworks explaining the power of these models and providing insights to their performance. In Cohen et al. (2016a), the authors considered convolutional networks with linear activations and product pooling layers, called convolutional arithmetic circuits (ConvACs), and argued for the expressiveness of deep networks using a tensor based analysis. This analysis has been extended to rectifier based convolutional networks via generalization of the tensor product Cohen & Shashua (2016a). In Cohen & Shashua (2016a), it was shown that ConvACs enjoy a greater expressive power than rectifier based models despite the popularity of rectifier based networks in practice. Indeed the empirical relevance of ConvAC was demonstrated through an architecture called SimNets Cohen et al. (2016b). In addition, the generative ConvAC of Sharir et al. (2016) achieved state of the art performance in classification of images with missing pixels. These results served as motivation for the works of Cohen & Shashua (2016b); Cohen et al. (2017); Levine et al. (2017); Sharir & Shashua (2017), where different aspects of ConvACs were studied from a theoretical perspective.

In Cohen & Shashua (2016b) the inductive bias introduced by pooling geometries was studied. Later, Levine et al. (2017) makes use of the quantum entanglement measure to analyze the inductive bias introduced by the correlations among the channels of ConvACs. Moreover, Sharir & Shashua (2017) generalizes the convolutional layer of ConvACs by allowing overlapping receptive fields, in other words permitting stride values lower than the convolution patch size. These locally overlapping connections led to an enhancement on the expressive capacity of ConvACs. The notion of inter-layer connectivity for ConvACs was addressed by Cohen et al. (2017) in the context of sequential data processing, such as audio and text related tasks. In that work, the expressive capabilities of interconnecting processing blocks from a sequence was studied. Nevertheless, these types of interconnections are related to the sequential nature of the problem and different from the ones used in ResNet, FractalNet and DenseNet.

In this work, we extend the tensor analysis framework of Cohen et al. (2016a) to obtain insightful knowledge about the effect of dense connections, from the kind used in DenseNets, FractalNet and ResNet, on the expressiveness of deep ConvACs. We study the expressive capabilities provided by different types of dense connections. Moreover, from these results we derive performance bounds and practical guidelines for selection of the hyperparameters of a deep ConvAC, such as layer widths and the topology of dense connections. These results serve as the first step into understanding dense connectivity in rectifier networks as well, since they can be further extended to include rectifier linear units, in the same spirit as the generalization of the tensor products done by Cohen & Shashua (2016a).

The remainder of this paper is organized as follows. In Section 2, we introduce the notation and basic concepts from tensor algebra. In Section 3, we present the tensor representation of ConvACs as introduced by Cohen et al. (2016a), and later in Section 4, we obtain tensor representations for densely connected ConvACs. In Section 5, performance bounds and design guidelines are derived for densely connected ConvACs.

2 PRELIMINARIES

The term *tensor* refers to a multi-dimensional array, where the *order* of the tensor corresponds to the number of indexes required to access one of its entries. For instance, a vector is a tensor of order 1 while a matrix is a tensor of order 2. In general a tensor $\mathcal{A}$ of order N requires N indexes $(d_1, \ldots, d_N)$ to access one of its elements. For the sake of notation, given $I \in \mathbb{N}$, we use the expression $[I]$ to denote the set $\{1, 2, \ldots, I\}$. In addition, the $(d_1, \ldots, d_N)$-th entry of a given tensor of order N and size $M_1 \times M_2 \times \cdots \times M_N$ is denoted as $\mathcal{A}_{d_1,\ldots,d_N}$, where $d_i \in [M_i]$ for all $i \in [N]$. Moreover, for the particular case of tensors of order N with symmetric sizes $M_1 = M_2 = \cdots = M_N = M$, we use $(\mathbb{R}^M)^{\otimes N}$ as shorthand notation for $\mathbb{R}^{M \times \cdots \times M}$. A crucial operator in tensor analysis is the tensor product $\otimes$, since it is necessary for defining the rank of a tensor. For two tensors $\mathcal{B} \in \mathbb{R}^{M_1 \times \cdots \times M_p}$ and $\mathcal{C} \in \mathbb{R}^{M_{p+1} \times \cdots \times M_{p+q}}$, the tensor product is defined such that $\mathcal{B} \otimes \mathcal{C} \in \mathbb{R}^{M_1 \times \cdots \times M_{p+q}}$ and $(\mathcal{B} \otimes \mathcal{C})_{d_1,\ldots,d_{p+q}} = \mathcal{B}_{d_1,\ldots,d_p} \mathcal{C}_{d_{p+1},\ldots,d_{p+q}}$ for all $(d_1, \ldots, d_{p+q})$. In tensor algebra, a tensor $\mathcal{A} \in \mathbb{R}^{M_1 \times M_2 \times \cdots \times M_N}$ is said to have rank 1 if it can be expressed as $\mathcal{A} = \mathbf{v}^{(1)} \otimes \cdots \otimes \mathbf{v}^{(N)}$, where $\mathbf{v}^{(i)} \in \mathbb{R}^{M_i}$ for all $i \in [N]$. Moreover, any tensor $\mathcal{A} \in \mathbb{R}^{M_1 \times M_2 \times \cdots \times M_N}$ can be expressed as a sum of rank-1 tensors, that is

$$\mathcal{A} = \sum_{z=1}^{Z} \mathbf{v}_z^{(1)} \otimes \cdots \otimes \mathbf{v}_z^{(N)} , \tag{1}$$

where $Z \in \mathbb{N}$ is sufficiently large and $\mathbf{v}_z^{(i)} \in \mathbb{R}^{M_i}$ for $i \in [N]$. Note that this statement is trivial for $Z = \prod_{i=1}^{N} M_i$. On the other hand, when Z is the minimum number such that (1) is satisfied, the rank of the tensor is defined to be $\mathrm{rank}(\mathcal{A}) = Z$ and (1) becomes equivalent to the well known CANDECOMP/PARAFAC (CP) decomposition of $\mathcal{A}$. Another operator, that is on the core of the former works of Cohen & Shashua (2016a); Cohen et al. (2016a); Levine et al. (2017), is the *matricization* operator. The operator $[\mathcal{A}]$ denotes the matricization of a tensor $\mathcal{A} \in \mathbb{R}^{M_1 \times \cdots \times M_N}$ of order N. This matricization of the tensor $\mathcal{A}$ re-orders its elements into a matrix $[\mathcal{A}] \in \mathbb{R}^{M_1 \cdot M_3 \cdots M_{N-1} \times M_2 \cdot M_4 \cdots M_N}$ with $\mathcal{A}^{y}_{d_1,\ldots,d_N}$ in the row $1 + \sum_{i=1}^{N/2} (d_{2i-1} - 1) \prod_{j=i+1}^{N/2} M_{2j-1}$ and column $1 + \sum_{i=1}^{N/2} (d_{2i-1} - 1) \prod_{j=i+1}^{N/2} M_{2j}$. This is operator is of great use since it enjoys properties such as $[\mathcal{A} \otimes \mathcal{B}] = [\mathcal{A}] \odot [\mathcal{B}]$ and $\mathrm{rank}(\mathcal{A}) \geq \mathrm{rank}([\mathcal{A}])$, where $\odot$ denotes the Kronecker

product of two matrices. Note that, since the Kronecker product is multiplicative in the rank, we have that $\text{rank}(\mathcal{A} \otimes \mathcal{B}) \geq \text{rank}([\mathcal{A} \otimes \mathcal{B}]) = \text{rank}([\mathcal{A}])\text{rank}([\mathcal{B}])$ which is a central property of this theoretical analysis framework.

3 Convolutional Arithmetic Circuits as Tensor Decompositions

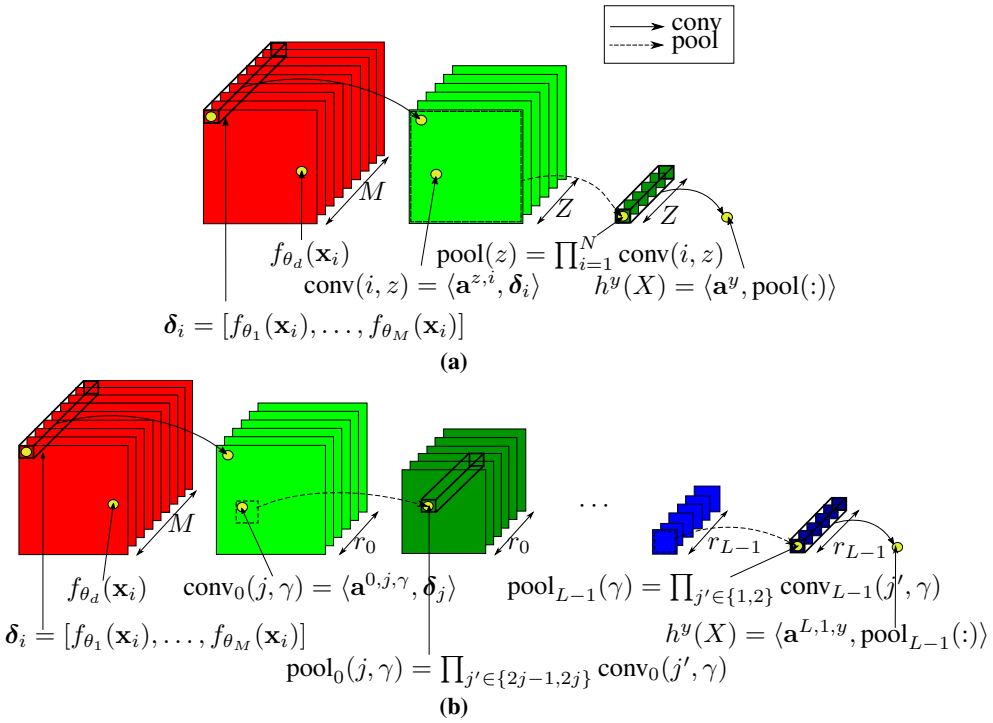

Figure 1: **(a)** Example of a shallow (i.e., $L = 1$) convolutional arithmetic circuit. **(b)** Example of a deep convolutional arithmetic circuit.

A ConvAC is a convolutional neural network that utilizes linear activation functions with product pooling, unlike most popular convolutional networks which make use of rectifier activations with max or average pooling. Moreover, the input of the network is modeled by $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_N) \in (\mathbb{R}^s)^N$, where $\mathbf{x}_i \in \mathbb{R}^s$ denotes the vectorization of the i-th patch of the input image. For this analysis, it is assumed that a set of M features is obtained from every patch, that is $f_{\theta_d}(\mathbf{x}_i) \in \mathbb{R}$ for all $i \in [N], d \in [M]$. These features are selected from a given parametric family $\mathcal{F} = \{f_\theta : \mathbb{R}^s \to \mathbb{R} : \theta \in \Theta\}$, such as Gaussian kernels, wavelet functions, or learned features. Then, to determine whether an input $\mathbf{X}$ belongs to a class belonging to the set $\mathcal{Y}$, the network evaluates the some score functions $h_y(\mathbf{X}) \in \mathbb{R}$ and decides for the class $y \in \mathcal{Y}$ such that

$$h_y(\mathbf{X}) = \max_{y \in \mathcal{Y}} h_y(\mathbf{X}) \,.$$

Using this formulation, in Figure 1**(a)** we observe an example of a single hidden layer ConvAC, while in Figure 1**(b)** we observe the general case of a deep arithmetic circuit of L layers. As shown by Cohen et al. (2016a), any score function of a ConvAC can be expressed as an homogeneous polynomial with degree N on the input features of the form

$$h_y(\mathbf{X}) = \sum_{d_1, \ldots, d_N = 1}^{M} \mathcal{A}^y_{d_1, \ldots, d_N} \prod_{i=1}^{N} f_{\theta_{d_i}}(\mathbf{x}_i) \,, \tag{2}$$

where $\mathcal{A}^y_{d_1, \ldots, d_N} \in \mathbb{R}$ are the polynomial coefficients stored in the grid-tensor $\mathcal{A}^y \in (\mathbb{R}^M)^{\otimes^N}$. In other words, a score function $h_y(\mathbf{X})$ is a polynomial of MN variables $f_{\theta_d}(\mathbf{x}_i) \in \mathbb{R}$ for all $i \in [N], d \in [M]$, degree N, and M^N polynomial coefficients stored in the grid-tensor $\mathcal{A}^y$.

For the special case of a shallow ConvAC with 1×1 convolutions and Z hidden units[1], shown in Figure 1**(a)**, the score functions are computed from the weight vectors $\mathbf{a}^{z,i} \triangleq [a_1^{z,i}, \ldots, a_M^{z,i}]^{\mathrm{T}} \in \mathbb{R}^M$ and $\mathbf{a}^y \triangleq [a_1^y, \ldots, a_Z^y]^{\mathrm{T}} \in \mathbb{R}^Z$ for all $i \in [N]$ and $z \in [Z]$. This leads to the score function

$$h_y(\mathbf{X}) = \langle \mathbf{a}^y, \mathrm{pool}(:) \rangle = \sum_{z=1}^{Z} a_z^y \, \mathrm{pool}(z) = \sum_{z=1}^{Z} a_z^y \prod_{i=1}^{N} \sum_{d=1}^{M} a_d^{z,i} f_{\theta_d}(\mathbf{x}_i) \, . \tag{3}$$

The first step of the tensor analysis framework is to obtain an expression (in terms of the network parameters a_z^y and $a_d^{z,i}$) of the grid-tensor $\mathcal{A}^y$ that represents this concrete network architecture. In other words, obtaining the expression for $\mathcal{A}^y$ that transforms (2) into (3). This expression was already obtained in Cohen & Shashua (2016a) as

$$\mathcal{A}^y = \sum_{z=1}^{Z} a_z^y \mathbf{a}^{z,1} \otimes \mathbf{a}^{z,2} \otimes \cdots \otimes \mathbf{a}^{z,N} \, , \tag{4}$$

where $\otimes$ denotes the tensor product. Note that (4) is in the form of a standard CP decomposition of the grid tensor $\mathcal{A}^y$. This implies that the rank of $\mathcal{A}^y$ is bounded by $\mathrm{rank}(\mathcal{A}^y) \leq Z$. Moreover, the obtained results where generalized in Cohen et al. (2016a) for the case of a deep ConvAC with size-2 pooling windows[2], thus $L = \log_2 N$ hidden layers as shown in Figure 1**(b)**, leading to a grid-tensor given by the hierarchical tensor decomposition

$$\begin{aligned}
\phi^{1,j,\gamma} &= \sum_{\alpha=1}^{r_0} a_\alpha^{1,j,\gamma} \mathbf{a}^{0,2j-1,\alpha} \otimes \mathbf{a}^{0,2j,\alpha} \\
&\vdots \\
\phi^{l,j,\gamma} &= \sum_{\alpha=1}^{r_{l-1}} a_\alpha^{l,j,\gamma} \phi^{l-1,2j-1,\alpha} \otimes \phi^{l-1,2j,\alpha} \\
&\vdots \\
\mathcal{A}^y = \phi^{L,1,1} &= \sum_{\alpha=1}^{r_{L-1}} a_\alpha^{L,1,y} \phi^{L-1,1,\alpha} \otimes \phi^{L-1,2,\alpha} \, ,
\end{aligned} \tag{5}$$

where $r_0, \ldots, r_{L-1} \in \mathbb{N}$ are the number of channels in the hidden layers, $\{\mathbf{a}^{0,j,\gamma} \in \mathbb{R}^M\}_{j \in [N], \gamma \in [r_0]}$ are the weights in the first hidden convolutions, $\{\mathbf{a}^{l,j,\gamma} \in \mathbb{R}^M\}_{j \in [N/2^l], \gamma \in [r_l]}$ are the weights of the hidden layers, and $\mathbf{a}^{L,1,y} \in \mathbb{R}^{r_{L-1}}$ stores the weights corresponding to the output y in the output layer.

4 Densely Connected Arithmetic Circuits

The recent empirical success of densely connected networks (DenseNets), presented by Huang et al. (2016), has served as motivation for our theoretical analysis on dense connectivity. Dense connectivity in a convolutional neural network refers to the case when a number $k \in \mathbb{N}$ (known as growth rate) of previous layers serve as input of the forthcoming layer. More precisely, in Huang et al. (2016), a DenseNet performs this via concatenation along the feature dimension of the current layer inputs with the preceding layer features. Note that these feature must have compatible sizes along the spatial dimension for the concatenation to be possible. To address this issue, Huang et al. (2016) proposed to group blocks of the same spatial dimensions into a *dense block*, as shown in Figure 2. These dense blocks do not contain operations such as pooling, that alter the spatial dimensions of the input features. Moreover, in the DenseNet architecture the layers that perform the pooling operation are called *transition layers*, since they serve as transition between dense blocks. For example, in Figure 2 we depict a *dense block* of 4 layers with growth rate $k = 2$, followed by a transition layer.

[1] We must mention that the generalization to $w \times w$ convolutions is straightforward and was already covered by Cohen & Shashua (2016a).

[2] Note that the generalization to different pooling sizes is straight forward and was done by Cohen & Shashua (2016a).

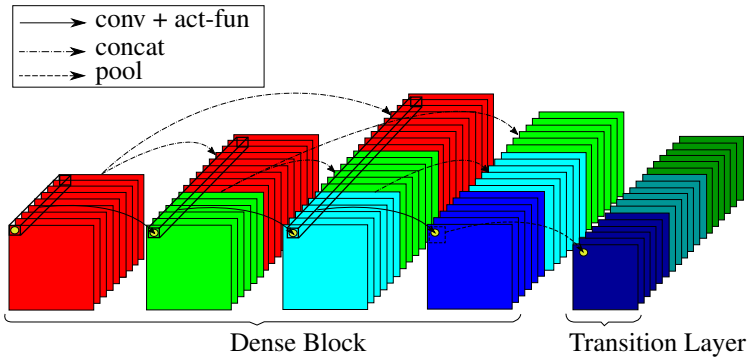

Figure 2: Example of a dense block of size 4 with growth rate $k = 2$.

In the original DenseNet these transition layers included one convolution layer before the pooling operation. Nevertheless, for this work we consider transition layers composed of only pooling operations. Note that this does not affect the generality of the model, since avoiding dense connections on the convolutional layer preceding the transition layer is equivalent to including a convolution in that transition layer[3].

In the case of ConvACs, any dense block of size greater than 1 can be represented as a dense block of size 1, since the activation function is the linear function (the non-linearity comes from the product pooling operator in the transition layer). Therefore, for ConvACs, it is only reasonable to analyze dense blocks of size 1. Note that, if we only allow dense connections between hidden layers within a dense block, a ConvAC is limited to a maximum growth rate of $k = 1$. In order to analyze the effect of broader connectivity we extend the concept of growth rate by allowing dense connections between dense blocks. With proper pooling, outputs of hidden layers belonging to different dense blocks can also be concatenated along the feature dimension. In the reminder of this paper we refer to the dense connections between hidden layers of the same block as *intra-block connections*, while the connections between hidden layers of different blocks as *inter-block connections*.

4.1 Dense Intra-block Connections

In this section we analyze the effect of intra-block connections. We first start by constructing a densely connected version of a single hidden layer ConvAC. The resulting network with growth rate $k = 1$ is shown in Figure 3**(a)**. In the same manner as in (3), this architecture leads to the score function

$$h_y(\mathbf{X}) = \sum_{z=1}^{Z} a_z^y \prod_{i=1}^{N} \sum_{d=1}^{M} a_d^{z,i} f_{\theta_d}(\mathbf{x}_i) + \sum_{z=Z+1}^{Z+M} a_z^y \prod_{i=1}^{N} f_{\theta_{z-Z}}(\mathbf{x}_i)\,. \tag{6}$$

Then, we present the following proposition regarding shallow ConvACs with dense connections of growth rate $k = 1$.

Proposition 1 *The network's function of a densely connected shallow ConvAC shown in* (6) *corresponds to the grid tensor*

$$\mathcal{A}^y = \sum_{z=1}^{Z} a_z^y \mathbf{a}^{z,1} \otimes \mathbf{a}^{z,2} \otimes \cdots \otimes \mathbf{a}^{z,N} + \underset{N}{\mathrm{Sdiag}} \left\{ a_{z+Z}^y \right\}_{z=1}^{M}\,, \tag{7}$$

where $\mathrm{Sdiag}_N \left\{ a_{z+Z}^y \right\}_{z=1}^{M} \in (\mathbb{R}^M)^{\otimes N}$ *denotes the super-diagonal tensor of order* N *with* $a_{Z+1}^y, \ldots, a_{Z+M}^y$ *in its diagonal.*

Proof See appendix B.1.

Note that the rank of this tensor is now bounded by $\mathrm{rank}(\mathcal{A}^y) \leq Z + M$ instead of Z, but adding these dense connections increases the number of parameters of the network from $MNZ + Z$ to

[3]This would effectively reduce the dense block size by 1.

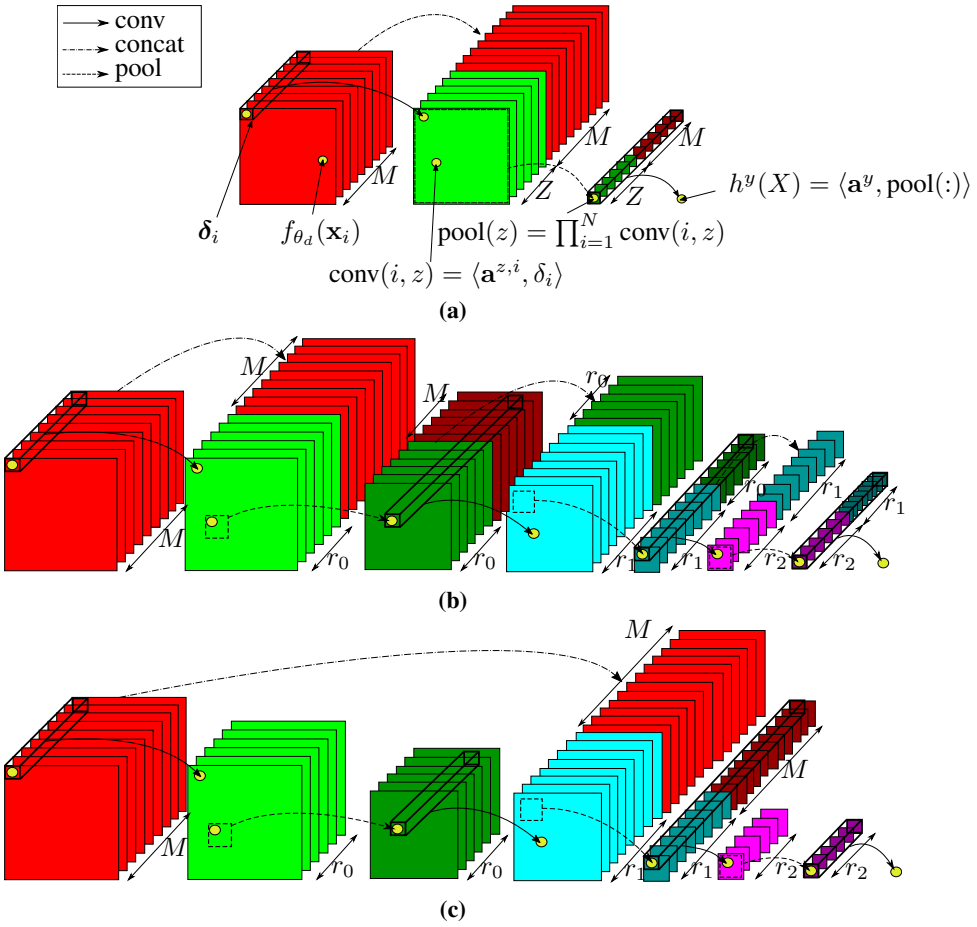

Figure 3: **(a)** Example of a shallow ($L = 1$) convolutional arithmetic circuit with one intra-block connection. **(b)** Example of a 3 layered ($L = 3$) convolutional arithmetic circuit with multiple intra-block connections. **(c)** Example of a 3 layered ($L = 3$) convolutional arithmetic circuit with one inter-block connection.

$MNZ+Z+M$. Then, for large values of N, there is no clear advantage on using dense connections on a shallow ConvAC. Nevertheless, in Section 5 we show that dense connections are capable of increasing the expressive power of deep ConvACs, specially for large values of N.

We now generalize the obtained results for the case of a L-layered dense arithmetic circuit, with growth rare $k = 1$, as the one in Figure 3**(b)**. Similarly to (5), the obtained grid tensor has the hierarchical decomposition given by

$$\begin{aligned}
\phi^{1,j,\gamma} &= \sum_{\alpha=1}^{r_0} a_{\alpha}^{1,j,\gamma} \mathbf{a}^{0,2j-1,\alpha} \otimes \mathbf{a}^{0,2j,\alpha} + \underset{2}{\text{Sdiag}} \left\{ a_{\alpha+r_0}^{1,j,\gamma} \right\}_{\alpha=1}^{M} \\
\phi^{2,j,\gamma} &= \sum_{\alpha=1}^{r_1} a_{\alpha}^{2,j,\gamma} \phi^{1,2j-1,\alpha} \otimes \phi^{1,2j,\alpha} + \underset{2^2}{\text{Sdiag}} \left\{ a_{\alpha+r_1}^{2,j,\gamma} \right\}_{\alpha=1}^{r_0} \qquad (8) \\
&\vdots \\
\mathcal{A}^y = \phi^{L,1,1} &= \sum_{\alpha=1}^{r_{L-1}} a_{\alpha}^{L,1,y} \phi^{L-1,1,\alpha} \otimes \phi^{L-1,2,\alpha} + \underset{2^L}{\text{Sdiag}} \left\{ a_{\alpha+r_{L-1}}^{L,j,\gamma} \right\}_{\alpha=1}^{r_{L-2}} . \qquad (9)
\end{aligned}$$

From this result we observe that inter block connections account for virtually increasing the width of the network's hidden layers from r_l to $\tilde{r}_l \triangleq r_l + r_{l-1}$ for all $l = 0, 1, \ldots, L-1$, where $r_{-1} \triangleq$

M. Note that this increased width comes at the expense of increasing the network's parameters. Moreover, in Section 5 we discuss whether increasing the network's width via intra block dense connections leads to an enhancement in its overall expressive power.

4.2 DENSE INTER-BLOCK CONNECTIONS

In this section we study broader connectivity via dense inter-block connections. As discussed in Section 4, proper pooling of the preceding features must take place before the concatenating them into the current layer. Since this type of connections have not been considered in the former DenseNets, we propose 3 possible ways of realizing such connections (via product, average, or max pooling). For a ConvAC with pooling window size w_{pool}, an inter block connection that connects block $l \in [L]$ with block $p \in [L]$ is said to be of jump length $L_{\text{jump}} \in [L-1]$ if $p = l + L_{\text{jump}}$. An example of an inter block connection of jump length $L_{\text{jump}} = 1$ can be seen in Figure 3**(c)**. To perform this inter block connections, the sizes along the spatial dimensions of preceding features must be reduced by $L_{\text{jump}} w_{\text{pool}}$, before concatenating them along the feature dimension of layer l. This spatial size reduction may be realized via pooling of the preceding features with window size $L_{\text{jump}} w_{\text{pool}}$. When using a pooling layer the size along the feature dimension remains unchanged. Moreover, the type of pooling employed (product, average, or maximum) affects the expressive potential of the resulting ConvAC. Furthermore, the following proposition addresses the effect that adding dense inter block connections, via average pooling, has on the network function of a ConvAC.

Proposition 2 *Adding inter block connections via average pooling of jump length $L_{jump} \geq 1$ to a standard ConvAC with grid-tensor $\mathcal{A}^y \in (\mathbb{R}^M)^{\otimes N}$ leads to a network function of the form*

$$h_y(\mathbf{X}) = \sum_{d_1,\ldots,d_N=1}^{M} \mathcal{A}^y_{d_1,\ldots,d_N} \prod_{i=1}^{N} f_{\theta_{d_i}}(\mathbf{x}_i) + g(\mathbf{X}) \,,$$

where $g(\mathbf{X})$ contains polynomial terms on $f_{\theta_d}(\mathbf{x}_i)$ for $d \in [M], i \in [N]$ of degree lower than N.

Remark 1 *This result is also valid when the connections are done by addition instead of concatenation, as it is done in ResNet and FractalNet.*

Proof See appendix B.2.

From this proposition we conclude that adding inter block connections average pooling does not alter the grid tensor $\mathcal{A}^y$, instead these connections account for extra polynomial terms of degree strictly less than N. Note that, for the special case where the input features belong to an exponential kernel family, such as $\mathcal{F} = \{f_\theta(\mathbf{x}) = e^{\theta^{\mathrm{T}}\mathbf{x}} : \mathbb{R}^s \to \mathbb{R} : \theta \in \Theta\}$ or $\mathcal{F} = \{f_\theta(\mathbf{x}) = e^{\|\theta - \mathbf{x}\|_p} : \mathbb{R}^s \to \mathbb{R} : \theta \in \Theta\}$ where $\|\cdot\|_p$ denotes the ℓ_p norm with $p \in \mathbb{N}$, the number of polynomial terms is equivalent to the number of exponential basis that the network function can realize. Therefore, the another valid measure of expressiveness is the number of polynomial terms a ConvAC is able to realize. Given a certain ConvAC topology, the number of polynomial terms can be computed inductively by expanding the polynomial products of every layer via generalized binomial expansions. Such an analysis is left for future contributions. Moreover, if we perform this connections via product poling, the features to be concatenated correspond to polynomial terms of the same order. This leads to a generalization of the intra-block connections from 4.1, leading to virtually increased widths $\tilde{r}_l \triangleq r_l + \sum_{q=1}^{L_{\text{jump}}} r_{l-1-q}$. Finally, we leave the analysis of inter-block connections via maximum pooling for future work and consider only product pooling inter-block connections in the remainder of this paper.

5 PRACTICAL IMPLICATIONS

For the sake of comparison, let us assume networks with hidden layer widths r_l decaying (or increasing) at an exponential rate of $\lambda \in \mathbb{R}$. Formally, this is $r_l = \lambda r_{l-1} \in \mathbb{N}$, thus $r_l = (\lambda)^l r$ for all $l = 0, 1, \ldots, L-1$, where $r \triangleq r_0$. To shorten the notation, we denote as (L, r, λ, k) to a ConvAC with of exponential width decay $\lambda \in \mathbb{R}$, length $L \in \mathbb{N}$, initial with $r \in \mathbb{N}$ and growth-rate $k \in \mathbb{N}$. A growth-rate of $k = 0$ refers to a standard ConvAC with no dense connections.

Definition 1 *Suppose that the weights of a (L, r, λ, k) ConvAC, with $L, k \in \mathbb{N}$ and $r, \lambda \in \mathbb{R}$, are randomly drawn according to some continuous non-vanishing distribution. Then, this (L, r, λ, k) ConvAC is said to have weak dense gain $G_w \in \mathbb{R}$ if, with probability $p > 0$, we obtain score functions that cannot be realized by a $(L, r', \lambda, 0)$ ConvAC with $r' < G_w r$. When $p = 1$, this (L, r, λ, k) ConvAC is said to have a strong dense gain $G_s = G_w \in \mathbb{R}$.*

Using this definition we present a bound for the weak dense gain G_w in the following theorem.

Theorem 5.1 *Given $M \in \mathbb{N}$, any a (L, r, λ, k) ConvAC with $L > 1, r \leq M, \lambda \leq 1, k > 0$ has a dense gain is bounden by $G_w \leq \frac{M}{\lambda r}$.*

Proof See appendix B.3.

This general bound may serve as guideline for tayloring M and the widths $r_0, \ldots, r_{L-1}$ such that we exploit the expressiveness added by dense connections.

Theorem 5.2 *For the particular case of theorem 5.1 when $k = 1$, the weak dense gain is bounded by $G_w \leq \min\left(1 + \frac{1}{\lambda}, \frac{M}{\lambda r}\right)$.*

Proof See appendix B.3.

Using this result, we able able to quantify the expressive gain provided by dense inter block connections. If a ConvAC has a dense gain $G_\mathrm{w} = (1 + \frac{1}{\lambda})$ that is already close to the general bound from Theorem 5.1 it is less encouraging to include broader dense connections, since it would increase the number of parameters of the model while there is no room for a significant expressive gain increase. In this scenario, connections as the ones in ResNet and FractalNet may result more beneficial since they do not increase the size of the model, while at the same time enhancing its trainability.

Theorem 5.3 *For the particular case of theorem 5.1 when $k = 1$, if $r \leq \frac{1}{1+\lambda}\sqrt{M}$, then the bound of theorem 5.2 is achieved with equality and strong dense gain $G_s = 1 + \frac{1}{\lambda}$.*

Proof See appendix B.3.

This last theorem shows that there exist a regime where this bounds can be achieved with strong dense gain. Whether this is true outside this regime is still an open question, since further knowledge about the rank of random tensors is limited. Moreover, these theorems does not consider the additional amount of parameters added by dense connections. We complete our analysis by addressing this issue in the following proposition.

Proposition 3 *Let $\Delta P_{dense} \in \mathbb{N}$ be the additional number of parameters that are added to a $(L, r, \lambda, 0)$ ConvAC when we introduce dense connections of growth-rate $k > 0$. In the same manner, let $\Delta P_{stand} \in \mathbb{N}$ be the number of parameters that are added to a $(L, r, \lambda, 0)$ ConvAC when we increase its initial width r by a factor $G \in \mathbb{R}$. Then the ratio between ΔP_{dense} and ΔP_{stand} is greater than*

$$\frac{\Delta P_{stand}}{\Delta P_{dense}} \geq \frac{(G-1)M}{r\sum_{q=1}^{k}\lambda^{-1-q}\sum_{l=1}^{L}(\frac{\lambda^2}{2})^l} + \frac{(G^2-1)}{\sum_{q=1}^{k}\lambda^{-q}}\,.$$

Proof See appendix B.4.

The factor G from this proposition directly relates to the dense gain of a ConvAC, thus this ratio may be used to decide whether is interesting to add dense connections to a model (we want this ratio to be as large as possible). Finally Theorems 5.1 and 5.2 directly bound this ratio, which give the practitioner a guideline to decide which connections (if any) should be added to a given model.

REFERENCES

Richard Caron and Tim Traynor. The zero set of a polynomial. *WSMR Report*, pp. 05–02, 2005.

Nadav Cohen and Amnon Shashua. Convolutional rectifier networks as generalized tensor decompositions. In Maria Florina Balcan and Kilian Q. Weinberger (eds.), *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 955–963, New York, New York, USA, 20–22 Jun 2016a. PMLR. URL http://proceedings.mlr.press/v48/cohenb16.html.

Nadav Cohen and Amnon Shashua. Inductive bias of deep convolutional networks through pooling geometry. *CoRR*, abs/1605.06743, 2016b. URL http://arxiv.org/abs/1605.06743.

Nadav Cohen, Or Sharir, and Amnon Shashua. On the expressive power of deep learning: A tensor analysis. In Vitaly Feldman, Alexander Rakhlin, and Ohad Shamir (eds.), *29th Annual Conference on Learning Theory*, volume 49 of *Proceedings of Machine Learning Research*, pp. 698–728, Columbia University, New York, New York, USA, 23–26 Jun 2016a. PMLR. URL http://proceedings.mlr.press/v49/cohen16.html.

Nadav Cohen, Or Sharir, and Amnon Shashua. Deep simnets. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016b.

Nadav Cohen, Ronen Tamari, and Amnon Shashua. Boosting dilated convolutional networks with mixed tensor decompositions. *CoRR*, abs/1703.06846, 2017. URL http://arxiv.org/abs/1703.06846.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Gao Huang, Zhuang Liu, Kilian Q Weinberger, and Laurens van der Maaten. Densely connected convolutional networks. *arXiv preprint arXiv:1608.06993*, 2016.

Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Fractalnet: Ultra-deep neural networks without residuals. *arXiv preprint arXiv:1605.07648*, 2016.

Yoav Levine, David Yakira, Nadav Cohen, and Amnon Shashua. Deep learning and quantum entanglement: Fundamental connections with implications to network design. *CoRR*, abs/1704.01552, 2017. URL http://arxiv.org/abs/1704.01552.

Or Sharir and Amnon Shashua. On the expressive power of overlapping operations of deep networks. *arXiv preprint arXiv:1703.02065*, 2017.

Or Sharir, Ronen Tamari, Nadav Cohen, and Amnon Shashua. Tractable generative convolutional arithmetic circuits. *CoRR*, abs/1610.04167, 2016. URL http://arxiv.org/abs/1610.04167.

A PRELIMINARY LEMMAS

Lemma 1 *Given $Z \in \mathbb{N}$, let $\mathcal{A} \in (\mathbb{R}^M)^{\otimes^P}$ be a random tensor of even order $P \geq 2$ such that*

$$\mathcal{A} = \sum_{z=1}^{Z} \mathbf{a}_z^{(1)} \otimes \cdots \otimes \mathbf{a}_z^{(P)} ,$$

where $\mathbf{a}_z^{(k)} \in \mathbb{R}^M$ are randomly drawn from a non-vanishing continuous distribution for all $k \in [P]$ and $z \in [Z]$. Then, if $Z \leq M^{P/2}$ we have that $\operatorname{rank}(\mathcal{A}) = \operatorname{rank}([\mathcal{A}]) = Z$ with probability 1. This lemma also holds when for a subset $\mathcal{Z} \subseteq [Z]$ we have that $\mathbf{a}_z^{(k)} = a_z \mathbf{e}_z \in \mathbb{R}^M$ for all $z \in \mathcal{Z}$, where $a_z \in \mathbb{R}$ are randomly drawn from a non-vanishing continuous distribution.

Proof Using the definition of the matricization operator, we get that the matricization $\mathcal{A}$ is

$$[\mathcal{A}] = \sum_{z=1}^{Z} (\underbrace{\mathbf{a}_z^{(1)} \odot \mathbf{a}_z^{(3)} \odot \cdots \odot \mathbf{a}_z^{(P-1)}}_{\tilde{\mathbf{a}}_z^{(\text{odd})}})(\underbrace{\mathbf{a}_z^{(2)} \odot \mathbf{a}_z^{(4)} \odot \cdots \odot \mathbf{a}_z^{(P)}}_{\tilde{\mathbf{a}}_z^{(\text{even})}})^{\mathrm{T}} \in \mathbb{R}^{M^{(P/2)} \times M^{(P/2)}} . \tag{10}$$

Note that, from this expression, it is straight forward to see that the rank of $[\mathcal{A}]$ is always less or equal than Z.

Let $\tilde{\mathcal{Z}} \subseteq [M^{P/2}]$ be the subset $\tilde{\mathcal{Z}} = \{\tilde{z} : \tilde{z} = \frac{M^{(P/2)}-1}{M-1}(z-1)+1 : z \in [Z]\}$ and $\mathbf{U} \in \mathbb{R}^{M^{(P/2)} \times M^{(P/2)}}$ be a permuted version of $[\mathcal{A}]$ such that the first Z rows of $\mathbf{U}$ correspond to the rows $\tilde{z} \in \tilde{\mathcal{Z}}$ of $[\mathcal{A}]$, and the first Z columns of $\mathbf{U}$ correspond to the columns $\tilde{z} \in \tilde{\mathcal{Z}}$ of $[\mathcal{A}]$. Since permuting the rows and the columns of a matrix does not alter its rank, we have that $\mathbf{U}$ has the same rank as $[\mathcal{A}]$. Now, let us partition $\mathbf{U}$ into blocks as

$$\mathbf{U} = \begin{bmatrix} \mathbf{P} & \mathbf{Q} \\ \mathbf{W} & \mathbf{Z} \end{bmatrix} ,$$

where $\mathbf{P}$ is of size Z-by-Z, and $\mathbf{Q}, \mathbf{W}, \mathbf{Z}$ have matching dimensions. Note that, if $\operatorname{rank}(\mathbf{P}) = Z$ then $\operatorname{rank}(\mathbf{U}) \geq Z$, which leads to $Z \leq \operatorname{rank}(\mathbf{U}) = \operatorname{rank}([\mathcal{A}]) \leq Z$, thus $\operatorname{rank}([\mathcal{A}]) = Z$. Therefore, it is sufficient to show that $\operatorname{rank}(\mathbf{P}) = Z$ with probability 1 to conclude this proof. To that end, let us define the mapping from $\mathbf{x} \in \mathbb{R}^{MPZ}$ to $\mathbf{P} = \mathbf{P}(\mathbf{x})$ as

$$\mathbf{x} \triangleq \left[\mathbf{a}_1^{(1)^{\mathrm{T}}}, \ldots, \mathbf{a}_Z^{(1)^{\mathrm{T}}}, \mathbf{a}_1^{(2)^{\mathrm{T}}}, \ldots, \mathbf{a}_Z^{(P)^{\mathrm{T}}} \right]^{\mathrm{T}} .$$

Note that this definition of $\mathbf{x}$ implies that $\mathbf{a}_z^{(i)} = \mathbf{a}_z^{(i)}(\mathbf{x})$ for all $z \in [Z]$ and $i \in [P]$. Therefore, since $[\mathcal{A}]$ is computed as in (10), we have that $[\mathcal{A}] = [\mathcal{A}](\mathbf{x})$, thus $\mathbf{Q} = \mathbf{Q}(\mathbf{x})$ and $\mathbf{P} = \mathbf{P}(\mathbf{x})$. Now, $\det \mathbf{P}(\mathbf{x})$ is a polynomial on $\mathbf{x}$, then it either vanishes in a set of measure zero or its the zero-polynomial (see Caron & Traynor (2005)).

If we set $\mathbf{x}$ to be equal to some $\mathbf{x}_0 \in \mathbb{R}^{MPZ}$ such that $\mathbf{a}_z^{(i)} = \mathbf{e}_z$ for all $z \in [Z]$ and $i \in [P]$, we have that $\tilde{\mathbf{a}}_z^{(\text{odd})} = \tilde{\mathbf{a}}_z^{(\text{even})} = \mathbf{e}_{\tilde{z}} \in \mathbb{R}^{M^{P/2}}$ with $\tilde{z} \triangleq (\sum_{n=0}^{P/2-1} M^n)(z-1)+1 = \frac{M^{(P/2)}-1}{M-1}(z-1)+1$. Therefore, since $[\mathcal{A}]$ is calculated as in (10) and $\tilde{\mathbf{a}}_z^{(\text{odd})} \tilde{\mathbf{a}}_z^{(\text{even})^{\mathrm{T}}}$ is now a matrix with 1 on the entry $(\tilde{z}, \tilde{z})$ and zero elsewhere, we have that $[\mathcal{A}](\mathbf{x}_0)$ is a diagonal matrix with ones on the diagonal elements $\tilde{z} \in \tilde{\mathcal{Z}}$ and zero elsewhere. This leads to $\mathbf{P}(\mathbf{x}_0) = \mathbf{I}_Z$ which has a determinant $\det \mathbf{P}(\mathbf{x}_0) \neq 0$. Finally, since there exist $\mathbf{x}_0$ such that the polynomial $\det \mathbf{P}(\mathbf{x}_0)$ is not zero, we conclude that $\det \mathbf{P}(\mathbf{x})$ is not the zero-polynomial, which means that $\det \mathbf{P}(\mathbf{x}) \neq 0$ with probability 1, thus proving this lemma.

Lemma 2 *Let $\mathcal{A} \in (\mathbb{R}^M)^{\otimes^P}$ and $\mathcal{B} \in (\mathbb{R}^M)^{\otimes^P}$ be random tensors of even order $P \geq 2$ and $Z \in \mathbb{N}$ be tensors such that*

$$\mathcal{A} = \sum_{z=1}^{Z_1} \mathbf{a}_z^{(1)} \otimes \cdots \otimes \mathbf{a}_z^{(P)} , \qquad \mathcal{B} = \sum_{z=1}^{Z_2} \mathbf{b}_z^{(1)} \otimes \cdots \otimes \mathbf{b}_z^{(P)} ,$$

where $\mathbf{a}_z^{(i)} \in \mathbb{R}^M$ *and* $\mathbf{b}_z^{(i)} \in \mathbb{R}^M$ *are randomly drawn from a non-vanishing continuous distribution. Then, if* $Z_1 \leq M^{P/2}$ *and* $Z_2 \leq M^{P/2}$*, we have that* $\operatorname{rank}(\mathcal{A} \otimes \mathcal{B}) = Z_1 Z_2$ *with probability 1.*

Proof Let $\mathcal{C} \in (\mathbb{R}^M)^{\otimes 2P}$ be a random tensor defined as $\mathcal{C} = \mathcal{A} \otimes \mathcal{B}$. Therefore, we may express $\mathcal{C}$ as

$$\mathcal{C} = \sum_{z=1}^{Z} \sum_{q=1}^{Z} \mathbf{a}_q^{(1)} \otimes \cdots \otimes \mathbf{a}_q^{(P)} \otimes \mathbf{b}_z^{(1)} \otimes \cdots \otimes \mathbf{b}_z^{(P)} .$$

Then, we define rank-1 tensors $\mathcal{C}^{(q,z)}$ to be $\mathcal{C}^{(q,z)} = \mathbf{a}_q^{(1)} \otimes \cdots \otimes \mathbf{a}_q^{(P)} \otimes \mathbf{b}_z^{(1)} \otimes \cdots \otimes \mathbf{b}_z^{(P)}$ to get

$$\mathcal{C} = \sum_{q,z=1}^{Z} \mathcal{C}^{(q,z)} .$$

Since $\mathcal{C}$ is now expressed as a sum of $Z_1 Z_2$ rank-1 tensors, we have that $\operatorname{rank}(\mathcal{C}) \leq Z_1 Z_2$.

Since $Z_1 \leq M^{P/2}$ and $Z_2 \leq M^{P/2}$ we may use Lemma 1, this leads to $\operatorname{rank}([\mathcal{A}]) = Z_1$ and $\operatorname{rank}([\mathcal{B}]) = Z_2$ with probability 1. Finally we, use the properties of the Kronecker product to obtain the rank of the matricization $\mathcal{C}$ as $\operatorname{rank}([\mathcal{C}]) = \operatorname{rank}([\mathcal{A} \otimes \mathcal{B}]) = \operatorname{rank}([\mathcal{A}])\operatorname{rank}([\mathcal{B}])$, leading to

$$\operatorname{rank}([\mathcal{C}]) = Z_1 Z_2 \Rightarrow Z_1 Z_2 = \operatorname{rank}([\mathcal{C}]) \leq \operatorname{rank}(\mathcal{C}) \leq Z_1 Z_2 \Rightarrow \operatorname{rank}(\mathcal{C}) = Z_1 Z_2$$

with probability 1, thus proving the Lemma.

Lemma 3 *Let* $\mathcal{A} \in (\mathbb{R}^M)^{\otimes^P}$ *and* $\mathcal{B} \in (\mathbb{R}^M)^{\otimes^P}$ *be tensors of order* $P > 2$ *and* $Z \in \mathbb{N}$ *be tensors such that*

$$\mathcal{A} = \sum_{z=1}^{Z_1} \mathbf{a}_z^{(1)} \otimes \cdots \otimes \mathbf{a}_z^{(P)} , \qquad \mathcal{B} = \sum_{z=1}^{Z_2} \mathbf{b}_z^{(1)} \otimes \cdots \otimes \mathbf{b}_z^{(P)} ,$$

where $\mathbf{a}_z^{(i)} \in \mathbb{R}^M$ *and* $\mathbf{b}_z^{(i)} \in \mathbb{R}^M$ *are randomly drawn from a non-vanishing continuous distribution. Then, if* $Z_1 + Z_2 \leq M^{P/2}$*, we have that* $\operatorname{rank}(\mathcal{A} + \mathcal{B}) = Z_1 + Z_2$ *with probability 1.*

Proof Let $\mathcal{C} \in \mathbb{R}^{M \times \cdots \times M}$ be a tensor of order P defined as $\mathcal{C} = \mathcal{A} + \mathcal{B}$. Therefore, we may express $\mathcal{C}$ as

$$\mathcal{C} = \sum_{z=1}^{Z_1} \mathbf{a}_z^{(1)} \otimes \cdots \otimes \mathbf{a}_z^{(P)} + \sum_{z=1}^{Z_2} \mathbf{b}_z^{(1)} \otimes \cdots \otimes \mathbf{b}_z^{(P)} = \sum_{z=1}^{Z_1+Z_2} \mathbf{c}_z^{(1)} \otimes \cdots \otimes \mathbf{c}_z^{(P)} ,$$

where

$$\mathbf{c}_z^{(i)} \triangleq \begin{cases} \mathbf{a}_z^{(i)} & 0 < z \leq Z_1 \\ \mathbf{b}_z^{(i)} & Z_1 < z \leq Z_1 + Z_2 \end{cases} .$$

Since $Z_1 + Z_2 \leq M^{P/2}$ we may use Lemma 1, leading to $\operatorname{rank}(\mathcal{C}) = Z_1 + Z_2$ with probability 1, thus proving this Lemma.

Corollary 1 *Let* $\mathcal{A}, \mathcal{B}, \mathcal{C}$ *be tensors of the same size with ranks* $Z_A \triangleq \operatorname{rank}(\mathcal{A})$*,* $Z_B \triangleq \operatorname{rank}(\mathcal{B})$*, and* $Z_C \triangleq \operatorname{rank}(\mathcal{C})$*. Then, the following statements hold true.*

$$\operatorname{rank}(\mathcal{A} + \mathcal{B} + \mathcal{C}) = Z_A + Z_B + Z_C \Rightarrow \operatorname{rank}(\mathcal{A} + \mathcal{B}) = Z_A + Z_B$$
$$\operatorname{rank}((\mathcal{A} + \mathcal{B}) \otimes \mathcal{C}) = (Z_A + Z_B) Z_C \Rightarrow \operatorname{rank}(\mathcal{A} \otimes \mathcal{C}) = Z_A Z_C .$$

B DEFERRED PROOFS

B.1 PROOF PROPOSITION 1

Proof We reformulate this (6) to have the same form as (3). To that end we define $a_z^{d,i}$, for $z = Z+1,\dots,Z+M$, to be $a_d^{z,i} = 1$ if $z - Z = d$ and zero otherwise. This definition of $a_d^{z,i}$ leads to $\sum_{d=1}^{M} a_d^{z,i} f_{\theta_d}(\mathbf{x}_i) = f_{\theta_{z-Z}}(\mathbf{x}_i)$ for $z = Z+1,\dots,Z+M$. Using this relation we get

$$\begin{aligned} h_y(\mathbf{X}) &= \sum_{z=1}^{Z} a_z^y \prod_{i=1}^{N}\sum_{d=1}^{M} a_d^{z,i} f_{\theta_d}(\mathbf{x}_i) + \sum_{z=Z+1}^{Z+M} a_z^y \prod_{i=1}^{N}\sum_{d=1}^{M} a_d^{z,i} f_{\theta_d}(\mathbf{x}_i) \\ &= \sum_{z=1}^{Z+M} a_z^y \prod_{i=1}^{N}\sum_{d=1}^{M} a_d^{z,i} f_{\theta_d}(\mathbf{x}_i)\,, \end{aligned}$$

which has the same form as (3). Therefore, as done in (4), we obtain the grid tensor for this architecture as

$$\begin{aligned} \mathcal{A}^y &= \sum_{z=1}^{Z+M} a_z^y \mathbf{a}^{z,1}\otimes\mathbf{a}^{z,2}\otimes\cdots\otimes\mathbf{a}^{z,N} \\ &= \sum_{z=1}^{Z} a_z^y \mathbf{a}^{z,1}\otimes\mathbf{a}^{z,2}\otimes\cdots\otimes\mathbf{a}^{z,N} + \sum_{z=Z+1}^{Z+M} a_z^y \mathbf{a}^{z,1}\otimes\mathbf{a}^{z,2}\otimes\cdots\otimes\mathbf{a}^{z,N} \\ &= \sum_{z=1}^{Z} a_z^y \mathbf{a}^{z,1}\otimes\mathbf{a}^{z,2}\otimes\cdots\otimes\mathbf{a}^{z,N} + \sum_{z=Z+1}^{Z+M} a_z^y \mathbf{e}_{z-Z}\otimes\mathbf{e}_{z-Z}\otimes\cdots\otimes\mathbf{e}_{z-Z} \\ &= \sum_{z=1}^{Z} a_z^y \mathbf{a}^{z,1}\otimes\mathbf{a}^{z,2}\otimes\cdots\otimes\mathbf{a}^{z,N} + \underset{N}{\mathrm{Sdiag}}\left\{a_{z+Z}^y\right\}_{z=1}^{M}\,, \end{aligned}$$

thus proving this proposition.

B.2 PROOF PROPOSITION 2

Proof Given $\mathbf{x} \triangleq [f_{\theta_1}(\mathbf{x}_1),\cdots,f_{\theta_M}(\mathbf{x}_1),f_{\theta_2}(\mathbf{x}_1),\cdots,f_{\theta_M}(\mathbf{x}_N)]^{\mathrm{T}} \in \mathbb{R}^{MN}$, the output of the l-th layer of a $(L,r,\lambda,0)$ ConvAC can be stored into the vectors of mappings $\boldsymbol{\delta}^{l,j}(\mathbf{x}) \triangleq [\delta_1^{l,j}(\mathbf{x}),\dots,\delta_{r_l}^{l,j}(\mathbf{x})]^{\mathrm{T}} \in \mathbb{R}^{r_l}$ for $j \in [N/2^l]$ and $l \in [L]$. Moreover, since the entries of these vectors are the result of $l-1$ convolution-pooling layers with product pooling of window size 2, all the mappings $\delta_1^{l,j}(\mathbf{x})$ can be expressed as a sum of polynomial terms on $\mathbf{x}$ of degree 2^l.

Now, let the coefficient vectors $\mathbf{a}^{l,j,\gamma} \triangleq [a_1^{l,j,\gamma},\dots,a_{r_l}^{l,j,\gamma}]^{\mathrm{T}} \in \mathbb{R}^{r_l}$ for $j \in [N/2^l]$ and $\gamma \in [r_{l+1}]$, be the weight vectors for the convolution of the l-th layer. To shorten the notation we use $\langle \mathbf{a}^{l,j,\gamma}, \boldsymbol{\delta}^{l,j}\rangle = \sum_{d=1}^{r_l} a_d^{l,j,\gamma}\delta_d^{l,j}$ as shorthand for the convolution between these vectors. Then, the outputs the the layer l of this ConvAC are given by $\boldsymbol{\delta}^{l+1,j} \in \mathbb{R}^{r_{l+1}}$ with $\delta_\gamma^{l+1,j} = \langle \mathbf{a}^{l,2j-1,\gamma}, \boldsymbol{\delta}^{l,2j-1}\rangle\langle \mathbf{a}^{l,2j,\gamma}, \boldsymbol{\delta}^{l,2j}\rangle$ for $j \in [N/2^{l+1}], \gamma \in [r_{l+1}]$. If we recursively calculate these out vectors up to the L-th layer we obtain the score functions $h_{\text{stand}}^y(\mathbf{x}) \triangleq \boldsymbol{\delta}^{L,1}(\mathbf{x}) = \delta_1^{L,1}(\mathbf{x}) \in \mathbb{R}$.

We now consider the effecct of adding dense connections via average pooling from some $k \in \mathbb{N}$ preceding layers $l-1,\dots,l-k$. To this end, let $\tilde{r}_l = \sum_{q=1}^{k} r_{l-q}$ be the total size along the feature dimesnion of the vectors to be concatenated. In addition, let $\boldsymbol{\omega}^{l,j}(\mathbf{x}) \triangleq [\omega_1^{l,j}(\mathbf{x}),\dots,\omega_{\tilde{r}_l}^{l,j}(\mathbf{x})]^{\mathrm{T}} \in \mathbb{R}^{\tilde{r}_l}$ be the vectors of mappings of the corresponding preceeding features at the layer l for $j \in [N/r_l]$. In order to compute the convolutions of this layer, an additional vector of coefficients is required as $\mathbf{b}^{l,j,\gamma} \triangleq [b_1^{l,j,\gamma},\dots,b_{\tilde{r}_l}^{l,j,\gamma}]^{\mathrm{T}} \in \mathbb{R}^{\tilde{r}_l}$. Then, the outputs of the l-th layer of this (L,r,λ,k) ConvAC are the denoted as the vectors $\tilde{\boldsymbol{\delta}}^{l,j}(\mathbf{x}) \triangleq [\tilde{\delta}_1^{l,j}(\mathbf{x}),\dots,\tilde{\delta}_{r_{l+1}}^{l,j}(\mathbf{x})]^{\mathrm{T}} \in \mathbb{R}^{r_{l+1}}$ for $j \in [N/2^{l+1}]$ where

$$\tilde{\boldsymbol{\delta}}^{l+1,j} = \langle \begin{bmatrix}\mathbf{a}^{l,2j-1,\gamma}\\ \mathbf{b}^{l,2j-1,\gamma}\end{bmatrix}, \begin{bmatrix}\boldsymbol{\delta}^{l,2j-1}\\ \boldsymbol{\omega}^{l,2j-1}\end{bmatrix}\rangle\langle \begin{bmatrix}\mathbf{a}^{l,2j,\gamma}\\ \mathbf{b}^{l,2j,\gamma}\end{bmatrix}, \begin{bmatrix}\boldsymbol{\delta}^{l,2j}\\ \boldsymbol{\omega}^{l,2j}\end{bmatrix}\rangle\,.$$

From this expression it follows

$$\begin{aligned}\tilde{\boldsymbol{\delta}}^{l+1,j} &= (\langle \mathbf{a}^{l,2j-1,\gamma}, \boldsymbol{\delta}^{l,2j-1} \rangle + \langle \mathbf{b}^{l,2j-1,\gamma}, \boldsymbol{\omega}^{l,2j} \rangle)(\langle \mathbf{a}^{l,2j,\gamma}, \boldsymbol{\delta}^{l,2j} \rangle + \langle \mathbf{b}^{l,2j,\gamma}, \boldsymbol{\omega}^{l,2j} \rangle) \\ &= \boldsymbol{\delta}^{l+1,j} + \boldsymbol{\omega}^{l+1,j} \,,\end{aligned}$$

where

$$\begin{aligned}\boldsymbol{\omega}^{l+1,j} = &\langle \mathbf{b}^{l,2j-1,\gamma}, \boldsymbol{\omega}^{l,2j-1} \rangle \langle \mathbf{a}^{l,2j,\gamma}, \boldsymbol{\delta}^{l,2j} \rangle + \langle \mathbf{a}^{l,2j-1,\gamma}, \boldsymbol{\delta}^{l,2j-1} \rangle \langle \mathbf{b}^{l,2j,\gamma}, \boldsymbol{\omega}^{l,2j} \rangle \\ &+ \langle \mathbf{b}^{l,2j-1,\gamma}, \boldsymbol{\omega}^{l,2j-1} \rangle \langle \mathbf{b}^{l,2j,\gamma}, \boldsymbol{\omega}^{l,2j} \rangle \,.\end{aligned}$$

Note that the entries of $\boldsymbol{\omega}^{l,j}(\mathbf{x})$ are assumed to come from preceding layers with an appropiate average pooling. Since performing avergae pooling does not increase the degree of the polynomial terms involved (only product pooling does) and the jump length $L\text{jump}$ is at least 1, the entries of $\boldsymbol{\omega}^{l,j}(\mathbf{x})$ have at most polynomial degree 2^{l-1}, which is strictly less than the degree of the entries of $\boldsymbol{\delta}^{l,j}(\mathbf{x})$ (i.e., 2^l). Therefore, from the obtained expression of $\boldsymbol{\omega}^{l+1,j}$ we observe that it has polynomials withb degree no greater than $2^l + 2^{l-1}$, while the entries of $\boldsymbol{\delta}^{l+1,j}$ have a strictly higher degree of $2^l + 2^l = 2^{l+1}$.

Moreover, since $\langle \mathbf{a}^{l,j,\gamma}, \boldsymbol{\delta}^{l,j} + \boldsymbol{\omega}^{l,j} \rangle$ can be expressed as

$$\langle \mathbf{a}^{l,j,\gamma}, \boldsymbol{\delta}^{l,j} + \boldsymbol{\omega}^{l,j} \rangle = \langle \begin{bmatrix} \mathbf{a}^{l,j,\gamma} \\ \mathbf{a}^{l,j,\gamma} \end{bmatrix}, \begin{bmatrix} \boldsymbol{\delta}^{l,j} \\ \boldsymbol{\omega}^{l,j} \end{bmatrix} \rangle \tag{11}$$

we can make use of the obtained results in an unductive manner up to the L-th layer, thus leading to

$$h^y_{\text{dense}}(\mathbf{x}) = h^y_{\text{stand}}(\mathbf{x}) + g(\mathbf{x}) \,,$$

where $g(\mathbf{x})$ contains polynomial terms of $\mathbf{x}$ of order strictly less than N, thus proving this theorem. Note that this result also applies to additive and resudial connections, as de ones used in ResNet and FractalNet, since they can be expressed as in (11).

B.3 Proof of Theorems 5.1 to 5.3

Proof Given $M \in \mathbb{N}$, a $(L, r, \lambda, 0)$ ConvAC with $L > 1, r \leq M, \lambda \leq 1$ has a grid tensor $\mathcal{A}^y_{\text{stand}} \in (\mathbb{R}^M)^{\otimes N}$. For the forthcoming analysis let us assume $r_0 \leq M$. This assumption is done, so that we can write $\min\{r_0, M\} = r_0$, merely for notation purposes since we show that this does not affect the generality of the results. Using this assumption, we upper bound the rank of the grid tensor as

$$\begin{aligned}&\operatorname{rank}\left(\phi^{1,j,\gamma}\right) = \operatorname{rank}\left(\sum_{\alpha=1}^{r_0} a^{1,j,\gamma}_{\alpha} \mathbf{a}^{0,2j-1,\alpha} \otimes \mathbf{a}^{0,2j,\alpha}\right) \leq \min\{r_0, M\} = r_0 \\ &\operatorname{rank}\left(\phi^{2,j,\gamma}\right) \leq \sum_{\alpha=1}^{r_1} \operatorname{rank}\left(\phi^{1,2j-1,\alpha} \otimes \phi^{1,2j,\alpha}\right) \leq \sum_{\alpha=1}^{r_1} \operatorname{rank}\left(\phi^{1,2j-1,\alpha}\right) \operatorname{rank}\left(\phi^{1,2j,\alpha}\right) = r_1 r_0^2 \\ &\qquad\vdots \\ &\operatorname{rank}\left(\phi^{l,j,\gamma}\right) \leq \sum_{\alpha=1}^{r_{l-1}} \operatorname{rank}\left(\phi^{l-1,2j-1,\alpha} \otimes \phi^{l-1,2j,\alpha}\right) \leq \sum_{\alpha=1}^{r_{l-1}} \operatorname{rank}\left(\phi^{l-1,2j-1,\alpha}\right) \operatorname{rank}\left(\phi^{l-1,2j,\alpha}\right) \,.\end{aligned} \tag{12}$$

It was shown in Cohen & Shashua (2016a) that, when the weights are independently generated from some continuous distribution, we have that $\operatorname{rank}\left(\phi^{1,j,\gamma}\right) = \min\{r_0, M\}$ with probability 1. Note that, the bounds obtained for r_0 values greater than M is the same as for $r_0 = M$, thus implying that the assumption of $r_0 \leq M$ does not affect the generality of the results. Finally, by induction up to the L-th layer, we obtain a bound for the grid tensor rank as

$$\operatorname{rank}\left(\mathcal{A}^y_{\text{stand}}\right) = \operatorname{rank}\left(\phi^{L,1,1}\right) \leq \sum_{\alpha=1}^{r_{L-1}} \operatorname{rank}\left(\phi^{L-1,1,\alpha}\right) \operatorname{rank}\left(\phi^{L-1,2,\alpha}\right) = \prod_{l=0}^{L-1} r_l^{2^{L-l-1}} \,. \tag{13}$$

Since we assumed networks with hidden layer widths r_l decaying (or increasing) at an exponential rate of $\lambda \in \mathbb{R}$. Formally, this is $r_l = \lambda r_{l-1} \in \mathbb{N}$, thus $r_l = (\lambda)^l r$ for all $l = 0, 1, \ldots, L-1$, where

$r \triangleq r_0$. Therefore, we may simplify the obtained bound to

$$\operatorname{rank}\left(\mathcal{A}^y_{\text{stand}}\right) \leq \prod_{l=0}^{L-1} ((\lambda)^l r)^{2^{L-l-1}} = (\lambda)^{\sum_{l=0}^{L-1} l 2^{L-l-1}} r^{\sum_{l=0}^{L-1} 2^{L-l-1}} = (\lambda)^{2^L-1-L} r^{2^L-1}\,.$$

We this analysis by proving Theorem 5.1. To that end let $\mathcal{A}^y_{\text{dense}}$ be the grid tensor of a dense (L, r, λ, k) ConvAC with $k > 0$, while $\mathcal{A}^y_{\text{stand}}$ is the grid tensor of a $(L, r', \lambda, 0)$ ConvAC with $r' \in \mathbb{R}$. As discussed in Section 4, this dense version of the former $(L, r, \lambda, 0)$ ConvAC is equivalent to virtually increasing the widths of the ConvAC, which translates extra additive terms in the expressions from 12. Moreover, using corollary 1 we observe that, if the ranks of the tensors $\phi^{l,j,\gamma}$ are additive and multiplicative up to $\operatorname{rank}(\mathcal{A}^y_{\text{dense}}) > \operatorname{rank}(\mathcal{A}^y_{\text{stand}})$, so they are up to $\operatorname{rank}(\mathcal{A}^y_{\text{stand}})$. A weak dense gain value $G_{\text{w}} \in \mathbb{R}$ is achieved when there is a set of functions realized by the (L, r, λ, k) ConvAC that cannot be realized by $(L, r', \lambda, 0)$ ConvAC unless $r' = G_{\text{w}} r$. To bound this gain, let us assume the best case scenario where $\operatorname{rank}(\mathcal{A}^y_{\text{dense}})$ reaches the maximum possible rank, from the size of $\mathcal{A}^y_{\text{dense}}$ this can be at most $\operatorname{rank}(\mathcal{A}^y_{\text{dense}}) = M^{2^L-1}$. As discussed, would imply that the ranks of the tensors $\phi^{l,j,\gamma}$ are additive up to M^{2^L-1} for both ConvACs. Therefore, $\mathcal{A}^y_{\text{stand}}$ achieves its maximum rank given by $\operatorname{rank}\left(\mathcal{A}^y_{\text{stand}}\right) = (\lambda)^{2^L-1-L}(r')^{2^L-1}$. Then, a $(L, r', \lambda, 0)$ ConvAC is able to realize the functions of a (L, r, λ, k) ConvAC when $\operatorname{rank}\left(\mathcal{A}^y_{\text{stand}}\right) = \operatorname{rank}\left(\mathcal{A}^y_{\text{dense}}\right)$, thus $(\lambda)^{2^L-1-L}(r')^{2^L-1} = M^{2^L-1}$. Finally, since $r' = G_{\text{w}} r$, this leads to a maximum value of

$$G_{\text{w}} = \frac{M}{\lambda r} \lambda^{\frac{L}{2^L-1}} \leq \frac{M}{\lambda r}\,,$$

which proves Theorem 5.1.

For Theorem 5.2 we use consider the particular case of $k = 1$, which yields a core tensor given by the hierarchical tensor decomposition from (9). We use the same assumption of $r_0 \leq M$ and define the virtually increased widths $\tilde{r}_l \triangleq r_l + r_{l-1} \in \mathbb{N}$ for $l = 1, \ldots, L-1$ and $\tilde{r}_0 \triangleq M$. This leads to

$$\operatorname{rank}\left(\phi^{1,j,\gamma}\right) = \operatorname{rank}\left(\sum_{\alpha=1}^{r_0+M} a^{1,j,\gamma}_{\alpha} \mathbf{a}^{0,2j-1,\alpha} \otimes \mathbf{a}^{0,2j,\alpha}\right) \leq \min\{r_0 + M, M\} = \tilde{r}_0$$

$$\operatorname{rank}\left(\phi^{2,j,\gamma}\right) \leq \sum_{\alpha=1}^{r_1+r_0} \operatorname{rank}\left(\phi^{1,2j-1,\alpha} \otimes \phi^{1,2j,\alpha}\right) \leq \sum_{\alpha=1}^{\tilde{r}_1} \operatorname{rank}\left(\phi^{1,2j-1,\alpha}\right) \operatorname{rank}\left(\phi^{1,2j,\alpha}\right) = \tilde{r}_1 \tilde{r}_0^2$$

$$\vdots$$

$$\operatorname{rank}\left(\phi^{l,j,\gamma}\right) \leq \sum_{\alpha=1}^{r_{l-1}+r_{l-2}} \operatorname{rank}\left(\phi^{l-1,2j-1,\alpha} \otimes \phi^{l-1,2j,\alpha}\right) \leq \sum_{\alpha=1}^{\tilde{r}_{l-1}} \operatorname{rank}\left(\phi^{l-1,2j-1,\alpha}\right) \operatorname{rank}\left(\phi^{l-1,2j,\alpha}\right)$$

and

$$\operatorname{rank}\left(\mathcal{A}^y_{\text{dens}}\right) = \operatorname{rank}\left(\phi^{L,1,1}\right) \leq \prod_{l=0}^{L-1} \tilde{r}_l^{2^{L-l-1}}\,. \tag{14}$$

Note that for $r_l = \lambda r_{l-1} \in \mathbb{N}$ ($\lambda \in \mathbb{R}$), we get virtually increased widths $\tilde{r}_l = (1+\lambda)^l r = \left(\lambda\left(1 + \frac{1}{\lambda}\right)\right)^l r$, for all $l = 0, 1, \ldots, L-1$, leading to

$$\begin{aligned}\operatorname{rank}\left(\mathcal{A}^y_{\text{dens}}\right) &\leq \prod_{l=0}^{L-1} \left((\lambda(1+1/\lambda))^l r\right)^{2^{L-l-1}} = (\lambda(1+1/\lambda))^{\sum_{l=0}^{L-1} l 2^{L-l-1}} r^{\sum_{l=0}^{L-1} 2^{L-l-1}} \\ &= (\lambda(1+1/\lambda))^{2^L-1-L} r^{2^L-1}\,.\end{aligned} \tag{15}$$

As in for the proof of Theorem 5.1, the maximum dense gain G_{w} is obtained when $\operatorname{rank}(\mathcal{A}^y_{\text{dense}})$ reaches the maximum possible rank. In this particular case, this corresponds to $\operatorname{rank}(\mathcal{A}^y_{\text{dense}}) = \min\left((\lambda(1+1/\lambda))^{2^L-1-L} r^{2^L-1}, M^{2^L-1}\right)$. Furthermore, for obtaining $\operatorname{rank}\left(\mathcal{A}^y_{\text{stand}}\right) = \operatorname{rank}\left(\mathcal{A}^y_{\text{dense}}\right)$ a gain of

$$G_{\text{w}} \leq \min\left(1 + \frac{1}{\lambda}, \frac{M}{\lambda r}\right)$$

is required, thus proving Theorem 5.2.

Finally, for proving Theorem 5.3 we show that this bound can be attained with probability 1. Note that this bound is achieved when the inequalities from (12) hold with equality for all $l \in [L]$. The first inequality of (12) holds when the tensors $\left(\phi^{l-1,2j-1,1} \otimes \phi^{l-1,2j,1}\right), \cdots, \left(\phi^{l-1,2j-1,r_{l-1}} \otimes \phi^{l-1,2j,r_{l-1}}\right)$ are additive on the rank. This can be proven to be true with probability 1 when

$$\sum_{\alpha=1}^{r_{l-1}} \operatorname{rank}\left(\phi^{l-1,2j-1,\alpha} \otimes \phi^{l-1,2j,\alpha}\right) \leq M^{2^{l-1}} \tag{16}$$

by applying Lemma 3. In the same manner, the second inequality of (12) holds when the tensor pairs $\left(\phi^{l-1,2j-1,\alpha}, \phi^{l-1,2j,\alpha}\right)$ are multiplicative in the tensor rank. We may use Lemma 2 to prove this is the case with probability 1 if we can bound

$$\operatorname{rank}\left(\phi^{l-1,2j-1,\alpha}\right) \leq M^{2^{l-2}}. \tag{17}$$

In summary, if equations (16) and (17) hold, we may use Lemmas 3 and 2 to prove that (12) reaches equality with probability 1, thus implying that (13) also reaches equality everywhere outside a set Lebesgue measure zero. It is straight forward to see that, if (16) holds, so does (17).

For the case of a network with exponential width decay λ and $r \leq \frac{1}{\lambda}\sqrt{M}$ we have that

$$\sum_{\alpha=1}^{r_{l-1}} \operatorname{rank}\left(\phi^{l-1,2j-1,\alpha} \otimes \phi^{l-1,2j,\alpha}\right) \leq (\lambda)^{2^l-1-l} r^{2^l-1} \leq (\lambda)^{2^l} r^{2^l} \leq (\lambda)^{2^l} \left(\frac{1}{\lambda}\sqrt{M}\right)^{2^l} = M^{2^{l-1}}, \tag{18}$$

thus (16) holds. Therefore, within this regime of $r \leq \frac{1}{\lambda}\sqrt{M}$, we may ensure that the tensor rank is additive and multiplicative with probability 1.

Now we apply the same reasoning for a densely connected arithmetic circuit of L layers and width decay λ such that $r \leq \frac{1}{1+\lambda}\sqrt{M} = \frac{1}{\lambda(1+1/\lambda)}\sqrt{M}$. In the same manner as for the standard ConvAC we bound

$$\begin{aligned}\sum_{\alpha=1}^{r_{l-1}} \operatorname{rank}\left(\phi^{l-1,2j-1,\alpha} \otimes \phi^{l-1,2j,\alpha}\right) &\leq \left(\lambda(1+1/\lambda)\right)^{2^l-1-l} r^{2^l-1} \leq \left(\lambda(1+1/\lambda)\right)^{2^l} r^{2^l} \\ &\leq \left(\lambda(1+1/\lambda)\right)^{2^l} \left(\frac{1}{\lambda(1+1/\lambda)}\sqrt{M}\right)^{2^l} = M^{2^{l-1}},\end{aligned}$$

which enables us to make use of Lemmas 3 and 2 to prove that (15) holds with equality everywhere except from a set of Lebesgue measure zero. Note that, for $r \leq \frac{1}{1+\lambda}\sqrt{M} < \frac{1}{\lambda}\sqrt{M}$, we have that both equations (13) and (15) reach equality with probability 1, thus proving Theorem 5.3.

B.4 Proof Proposition 3

Proof Let $P(L, r, \lambda, k) \in \mathbb{N}$ be the number of parameters of a (L, r, λ, k) ConvAC. A standard $(L, r, \lambda, 0)$ ConvAC is composed of the weights $\{\mathbf{a}^{0,j,\gamma} \in \mathbb{R}^M\}_{j\in[N],\gamma\in[r_0]}$ in the first hidden convolutions, $\{\mathbf{a}^{l,j,\gamma} \in \mathbb{R}^M\}_{j\in[N/2^l],\gamma\in[r_l]}$ in the hidden layers, and $\mathbf{a}^{L,1,y} \in \mathbb{R}^{r_{L-1}}$ in the weights corresponding to the output y in the output layer. Therefore, this ConvAC has a number of weights

$$P(L, r, \lambda, 0) = MNr_0 + \sum_{l=1}^{L} \frac{N}{2^l} r_l r_{l-1} + Y r_L$$

When adding dense connections of growth-rate k, we need additional weights for the convolution between the preceding layers and the current layer. Therefore, at the l-th layer we have an extra $\sum_{q=1}^{k} \frac{N}{2^l} r_l r_{l-1-q}$ weights, which leads to

$$P(L, r, \lambda, k) = MNr_0 + \sum_{l=1}^{L} \frac{N}{2^l} r_l r_{l-1} + \sum_{l=1}^{L}\sum_{q=1}^{k} \frac{N}{2^l} r_l r_{l-1-q} + Y r_L.$$

By definition, we have that $\Delta P_{\text{std}} = P(L, Gr, \lambda, 0) - P(L, Gr, \lambda, 0)$ and $\Delta P_{\text{dense}} = P(L, Gr, \lambda, k) - P(L, r, \lambda, 0)$, thus yielding

$$\begin{aligned}
\Delta P_{\text{std}} &= (G-1)(MNr_0 + Yr_L) + (G^2-1)\sum_{l=1}^{L}\frac{N}{2^l}r_l r_{l-1} \\
&= (G-1)(MNr + \lambda^L Yr) + (G^2-1)\sum_{l=1}^{L}\frac{N}{2^l}\lambda^{2l-1}r^2 \\
&= (G-1)(MN + \lambda^L Y)r + (G^2-1)N\lambda^{-1}r^2\sum_{l=1}^{L}(\frac{\lambda^2}{2})^l
\end{aligned}$$

and

$$\Delta P_{\text{dense}} = \sum_{q=1}^{k}\sum_{l=1}^{L}\frac{N}{2^l}r_l r_{l-1-q} = Nr^2\sum_{q=1}^{k}\lambda^{-1-q}\sum_{l=1}^{L}(\frac{\lambda^2}{2})^l\,.$$

Finnaly, we use these expressions to compute the ratio of interest as

$$\begin{aligned}
\frac{\Delta P_{\text{std}}}{\Delta P_{\text{dense}}} &= \frac{(G-1)(MN + \lambda^L Y)r + (G^2-1)N\lambda^{-1}r^2\sum_{l=1}^{L}(\frac{\lambda^2}{2})^l}{Nr^2\sum_{q=1}^{k}\lambda^{-1-q}\sum_{l=1}^{L}(\frac{\lambda^2}{2})^l} \\
&= \frac{(G-1)(MN + \lambda^L Y)}{Nr\sum_{q=1}^{k}\lambda^{-1-q}\sum_{l=1}^{L}(\frac{\lambda^2}{2})^l} + \frac{(G^2-1)}{\sum_{q=1}^{k}\lambda^{-1-q}} \geq \frac{(G-1)M}{r\sum_{q=1}^{k}\lambda^{-1-q}\sum_{l=1}^{L}(\frac{\lambda^2}{2})^l} + \frac{(G^2-1)}{\sum_{q=1}^{k}\lambda^{-q}}
\end{aligned}$$

which proves this proposition.