IDENTIFYING A SAMPLE COMPLEXITY GAP BETWEEN CONVOLUTIONAL AND FULLY CONNECTED NEURAL NETWORKS

Orfeas Liossatos
Supervisor: Thomas Weinberger
Communication Theory Laboratory (LTHC)
EPFL

***Abstract*—Why can convolutional neural networks generalise with less data than fully connected neural networks, when in many computer vision tasks, the latter may simulate the former? This report advances the hypothesis that the difference arises from the data distribution and choice of optimisation algorithm, by describing a natural binary image classification task based on the wave patterns formed by the two-dimensional inverse discrete Fourier transform. We empirically demonstrate that in the over-parameterised regime, when employing the optimiser Adam, a shallow fully connected network requires more samples to generalise than a max-pooling convolutional neural network, even when controlling for the number of parameters in both models. For square images with d pixels, the fully connected network needs on the order of $\Omega(d)$ samples to generalise while the convolutional neural network requires $O(1)$.** [1]

I. INTRODUCTION

If we require a model to classify images at a fixed test accuracy, how much data do we need relative to the image size? If we don't define a data distribution, then we must require successful learning even for the worst-case distribution. Vapnik-Chervonenkis (VC) theory asserts that if it is at all possible for an instance of the model architecture to reach said accuracy, then it should depend on the expressiveness of the architecture itself: its VC-dimension. But if one model architecture is a subset of the other, like how a fully connected neural networks (FCNNs) can simulate a convolutional neural network (CNNs) by zeroing-out many of its weights, then the difference in their performance must be a result of the underlying distribution and chosen optimisation algorithm.

[1]Python notebooks are available at https://github.com/orfeasliossatos/Semester-Project/blob/main/Tasks/Project.ipynb

Explanations of this performance difference have been made by exhibiting natural image classification tasks that induce a sample complexity gap as a function of the number of pixels in the image (input dimension d). For instance, Li et. al. (2020) [1] describe the task of determining whether the 2-norm of half of the pixels is greater than the rest, yielding a provable gap that is quadratic in d, when both the FCNN and CNN are trained with SGD, albeit on a highly simplified CNN architecture. We generalize this class of tasks to a "difference of p-norms" (*DOp*) where $p > 0$. In the same vein, Brutzkus et. al. (2021) [2] describe a task of determining whether an image contains a given "positive" or "negative" pattern in a sea of spurious patterns. They show the potential for a gap between a max-pooling CNN with non-overlapping convolution kernels trained with the non-standard optimisation algorithm "layer-wise gradient descent", and a one-hidden layer FCNN with leaky ReLU activations trained with SGD. For the CNN they prove an upper bound on the sample complexity that is linear in the kernel size k and constant in the input dimension: $\mathcal{O}(k)$. For the FCNN they prove a quadratic upper bound $O(d^2)$ but we note that no lower bound on the sample complexity was proven.

In this report, we consider a natural image classification task we name "noisy Fourier patterns" (*NFP*) based on the wave patterns formed by the two-dimensional inverse discrete Fourier transform, and train CNNs and FCNNs with realistic architectures up to a fixed accuracy on increasing image sizes with Adam, to observe the required number of samples. In section II we recall fundamental definitions from VC theory. In section III we describe the data distribution, models, training protocol, and sample complexity estimation method for both tasks *DOp* and *NFP*. In section IV we report our results and interpret the measurements observed in Figure 3.

II. Background

A. Preliminaries

We recall the definitions of probably approximately correct learning (PAC-learning), VC-dimension, and empirical risk minimisation as per Shalev-Shwartz (2014) [3], simultaneously introducing notation for different components of the learning environment.

Definition 1 (Agnostic PAC-Learnability)**.** Let $\mathcal{X}, \mathcal{Y}$ be input and output domains. For any $\epsilon, \delta \in (0,1)$, a hypothesis class (model architecture) $\mathcal{H} = \{h : \mathcal{X} \to \mathcal{Y}\}$ is agnostic (ϵ, δ)-*PAC-learnable* if there exists a learning algorithm $\mathcal{A}$ taking a finite set of training samples $S = \{(x_i, y_i)\}_{i=1}^m$ and returning a hypothesis $\mathcal{A}(\mathcal{S}) \in \mathcal{H}$, and a minimum sample size function $m_{\mathcal{H}}(\epsilon, \delta)$ such that for any distribution $\mathcal{D}$ over $(\mathcal{X}, \mathcal{Y})$, the true loss of the hypothesis $\mathcal{A}(\mathcal{S})$, is within a accuracy ϵ of the true loss $L_{\mathcal{D}}$ of the best hypothesis $h^* \in \mathcal{H}$ with probability at least $1 - \delta$ over the randomness of the sample $S \overset{\text{iid}}{\sim} \mathcal{D}^m$ for every $m \geq m_{\mathcal{H}}(\epsilon, \delta)$. Formally, $\mathbb{P}_{\mathcal{S}}\{L_{\mathcal{D}}(\mathcal{A}(\mathcal{S})) \leq L_{\mathcal{D}}(h^*) + \epsilon\} \geq 1 - \delta$.

For a given learning algorithm (hypothesis class + optimiser), we are interested in its *sample complexity*: the asymptotic behaviour of the minimum sample size function $m_{\mathcal{H}}(\epsilon, \delta)$ in terms of the input dimension d, supposing $\mathcal{X} \subseteq \mathbb{R}^d$. A core principle of VC theory is that one needs more samples to learn richer hypothesis classes. So we recall the definition of VC-dimension, which is a notion of hypothesis class expressiveness.

Definition 2 (VC-Dimension)**.** A hypothesis class $\mathcal{H}$ *shatters* a finite set $C \subset \mathcal{X}$ if the restriction of $\mathcal{H}$ to C is the set of all functions from C to $\{0,1\}$. The *VC-dimension* $\text{VCdim}(\mathcal{H})$ of the hypothesis class $\mathcal{H}$ is the maximal size of a set $C \subset \mathcal{X}$ that can be shattered by $\mathcal{H}$.

The true distribution $\mathcal{D}$ is unknown to the learning algorithm $\mathcal{A}$ which only sees a finite set of samples $S \overset{\text{iid}}{\sim} \mathcal{D}^m$. The performance of a hypothesis h can still be tracked with an empirical loss $L_{\mathcal{S}}(h)$ that is an unbiased estimator of the true loss $L_{\mathcal{D}}(h)$. Learning algorithms that can minimise the empirical loss are often theoretically useful.

Definition 3 (Expected Risk Minimiser)**.** An expected risk minimising algorithm $\text{ERM}_{\mathcal{H}}$ outputs a hypothesis in $\mathcal{H}$ that minimises the empirical loss over a set of training samples S. Formally, $\text{ERM}_{\mathcal{H}}(\mathcal{S}) = \arg\min_{h \in \mathcal{H}} L_{\mathcal{S}}(h)$.

B. Sample Complexity Bounds from VC Theory

Sample complexity bounds resulting from VC theory are typically conservative, as the learning algorithm is required to learn any distribution $\mathcal{D}$ over $(\mathcal{X}, \mathcal{Y})$, whereas in many applications we typically want to know the sample complexity for particular classes of distributions. Nevertheless, for binary classification tasks with $\mathcal{Y} = \{0,1\}$, Blumer et. al. (1989) [4] showed that for a sample size of at least

$$m_{\mathcal{H}}(\epsilon, \delta) = \Omega(\max\{\frac{1}{\epsilon}\log\frac{1}{\delta}, \frac{1}{\epsilon}\text{VCdim}(\mathcal{H})\log\frac{1}{\epsilon}\}),$$

any expected risk minimiser that achieves zero empirical loss will (ϵ, δ)-PAC-learn the hypothesis class. More recently, Hanneke (2016) [5] showed in the realisable case $L_{\mathcal{D}}(h^*) = 0$ that there is an optimal algorithm based on a majority vote of expected risk minimisers that (ϵ,δ)-PAC-learns with

$$m_{\mathcal{H}}(\epsilon, \delta) = \Theta(\frac{1}{\epsilon}(\text{VCdim}(\mathcal{H}) + \log\frac{1}{\delta})).$$

Furthermore, the VC-dimension of various neural network architectures have been proven.

1) FCNNs: Harvey et. al. (2017) [6] derive an upper bound on the VC-dimension of a L-layer fully connected neural network with W parameters and ReLU activations: $\text{VCdim}(\text{FCNN}) = \mathcal{O}(WL\log W)$. When the network has only a single hidden layer with a bounded number of neurons, then the number of parameters of the network scales linearly with the input dimension d, so $\text{VCdim}(\text{FCNN}) = \mathcal{O}(d\log d)$.

2) CNNs: Brutzkus et. al. (2021) [2] find a lower bound for the VC-dimension of CNNs with a fixed number of filters, non-overlapping convolutions, a global max-pooling layer, and a final fully connected layer with ReLU activations. $\text{VCdim}(\text{CNN}) \geq 2^d$. [2]

C. Sample Complexity Gaps

The sample complexity bounds from VC theory are conservative since the learning algorithm $\mathcal{A}$ is meant to (ϵ, δ)-PAC-learn any distribution $\mathcal{D}$ over ($\mathcal{X}$,$\mathcal{Y}$). Therefore, when we restrict in advance the set of possible distributions to a class $\mathcal{P}$, the sample complexity will generally decrease. Although the sample complexity bounds in terms of VC-dimension are informative for the optimal algorithm, we are interested in ranking particular learning algorithms (hypothesis class + optimiser) in terms of their sample complexities. When we combine these two facts, we may observe sample complexity *gaps* between different kinds of learning algorithms.

[2]They shatter a set of size $2^{\frac{n}{k}-1}$ where k is the kernel size.

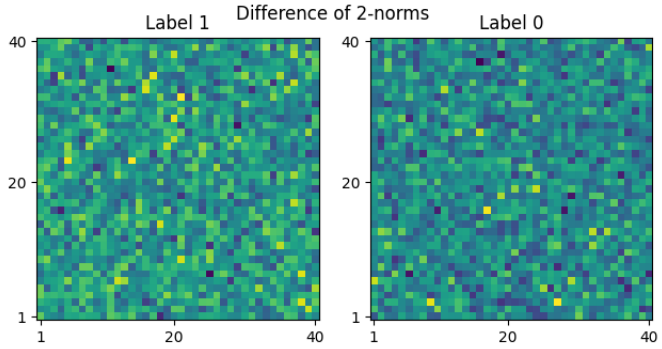

Fig. 1: Samples from DO2, with input dimension $d = 40^2$.

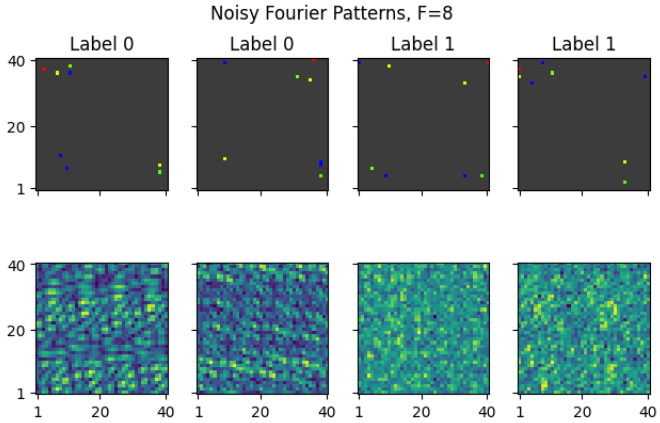

Fig. 2: Samples from NFP with 8 different complex frequencies with input dimension $d = 40^2$, represented in the top row with color corresponding to the complex angle. The lowest frequencies increase from the top-left corner and hence wrap around to the other sides. The bottom row is the corresponding L1-normalised pixel-wise modulus of the IFDT, with additive noise in the case of label 1.

Definition 4 (Sample Complexity Gap)**.** Let there be two learning algorithms $\mathcal{A}_1$ and $\mathcal{A}_2$. Fix a class of data distributions $\mathcal{P}$ over $\mathcal{X} \subset \mathbb{R}^d$ and $\mathcal{Y}$. We observe a *sample complexity gap* between $\mathcal{A}_1$ and $\mathcal{A}_2$ if $m_{\mathcal{H}_1} = \Omega(f(d))$ and $m_{\mathcal{H}_2} = \mathcal{O}(g(d))$, where $g = o(f)$ with f, g being non-decreasing functions of the input dimension d.

Such hard sample complexity gaps have been shown in Li et. al. (2020) [1]. For a simple pattern-recognition task, any permutation-equivariant learning algorithm (that is, one where permuting the data axes doesn't affect the output hypothesis such as simple FCNNs trained with Adam) needs at least $\Omega(d)$ samples, while the empirical risk minimizer ERM_{CNN} requires only $O(1)$ samples.

III. Models & Methods

We fix a test accuracy goal of $\epsilon = 80\%$ and compare two models III-B on two binary image classification tasks III-A across multiple image sizes and fit monomials to the means of recorded number of samples to obtain an estimate of the sample complexity.

A. Tasks

We slightly generalize the difference of 2-norms task defined in Li et. al. (2020) [1], and define our own noisy Fourier pattern task.

1) Difference of p-norms (DOp): An input is an entry-wise normal square image $\mathcal{X} \ni X \sim \mathcal{N}(0,1)^d$ with d a square number. Furthermore for $p > 0$, define the labelling function $h_p(X) = \mathbb{1}[\sum_{i=1}^{d/2} x_i^p > \sum_{i=d/2+1}^{d} x_i^p]$. Then $Y = h_p(X)$ is the corresponding label in $\{0, 1\}$. Some samples can be seen in Figure 1. This task is natural in the sense that it can be a specific bright pattern detector, depending on the indexing of the pixels x_i.

2) Noisy Fourier Patterns (DFP): In a blank square image $\hat{X}$ with d pixels, assign random numbers on the complex unit circle to $F = 8$ different random points in a low-pass box of sidelength $\sqrt{d}/2$. Compute the two-dimensional inverse discrete Fourier transform with pixel-wise modulus $|IDFT(\hat{X})|$ and standardise the pixel values to mean 0 and variance 1. Call this M. On the side, generate a label $Y \sim \text{Ber}(0.5)$ in $\{0, 1\}$. If $Y = 1$, obfuscate M with noise $Z \sim \mathcal{N}(0,1)^d$. Finally, normalise and rescale as follows.

$$X = L^2 \frac{M + YZ}{(||M + YZ||_1 + 10^{-6})}$$

A few samples can be seen in Figure 2. We claim that this resembles realistic computer vision tasks, as important features of a photo such as the face, ears, and eyes could correspond to low frequency components of the image's Fourier transform, and features such as face texture and individual strands of hair could correspond to high frequency components, represented by the added noise.

B. Models

We compare a max-pooling CNN to a FCNN with a single hidden layer. We match the FCNN's number of parameters to that of the CNN by adjusting the number of neurons in the hidden layer. The final outputs of the models are run through a logistic function σ such that

the outputs fall within the range $(0, 1)$. We are working in an over-parameterised regime because the number of output channels of the CNN is large enough to fit the training data with approximately zero training loss over a wide range of image sizes. The models are defined as follows:

a) CNN: A convolutional layer with one in-channel and $C = 1000$ out-channels with kernel size $k = 9$, stride 1, and padding, followed by ReLU activations and a max-pooling layer with non-overlapping kernels of size 4, and finally a fully connected layer with $Cd/4$ inputs and one output. It can therefore be noted that the total number of parameters $p_{\text{CNN}} = CK + Cd/4$ increases linearly with the image size.

b) FCNN: A fully connected neural network with one hidden layer with Q neurons and ReLU activations. We note that the number of parameters is $p_{\text{FCNN}} = dQ + Q$. In order to match p_{CNN} to p_{FCNN} as close as possible, we let $Q = \max\{1, \lfloor \frac{p_{\text{CNN}}}{(d+1)} \rfloor\}$.

Our choice of parameter count stands in contrast to the model architectures proposed in Li et. al. (2020) [1], where the $\mathcal{O}(1)$ upper bound is achieved by a CNN with just two output channels, yielding a VC dimension of 3. Instead, we work with a FCNN and CNN with more realistic capacity, and impose an extra fairness constraint of matching the number of parameters in both models.

C. Training and Testing

We train with Adam against a Binary Cross Entropy Loss function, with image batches of size $B = 64$ until the model reaches test loss of 80%. The learning rates η were chosen manually to avoid jumping or slow convergence, and are summarised in Table I. In order to strike a balance between total run-time and measurement accuracy, we adopted the following training protocol.

1) When the model obtains a training loss of 0.5, we test the model against a set of 1000 fixed image-label pairs and compute the test accuracy α.
2) If the obtained accuracy surpasses the requirement $\alpha > \epsilon$, then stop and record the total number of samples. Otherwise, lower the training loss goal by 0.025 and return to 1).

Of course, since batch training is discrete and the sample generation process is random, it can happen that the model overshoots the test-loss goal of ϵ. However, in practice it never appeared to overshoot by a large margin, and we perform the experiment 5 times to average out randomness.

TABLE I: Learning rates

η	**DO2**	**NFP**
CNN	0.0001	0.00005
FCNN	0.001	0.005

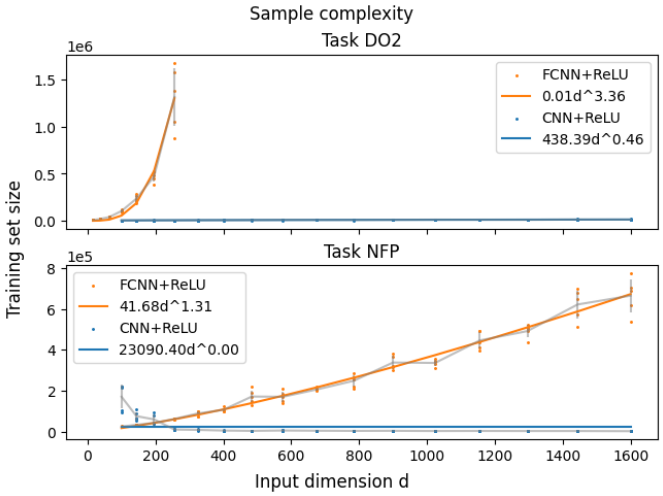

Fig. 3: Sample complexity estimates for FCNNs and CNNs with ReLU activations trained on the difference of 2-norms (DO2) and noisy Fourier patterns (NFP) tasks with Adam until at least 80% test accuracy on a set of 1000 image-label pairs. For each input dimension , five estimates were taken. Vertical error bars (grey) are the standard deviations in the estimations of the required training set size. The number of required training samples explodes in DO2 with FCNNs, so we compensate by performing the experiment over a reduced range of 4^2 to 16^2.

D. Degree Estimation

We train on a range of even image widths $\sqrt{d} \in \{10, 12, 14 \ldots, 40\}$. The image sizes are even because the square max-pooling kernels of the CNN are of width 2, so the parameter count only increases when the image width increases by 2. Including every image width would therefore result in a jagged graph. We perform the sample complexity estimation a total of 5 times per model and task, and then perform a nonlinear least-squares regression to fit a monomial of the form ax^b with $a, b \in \mathbb{R}_{\geq 0}$ in order to obtain estimates $\Omega(d^b)$. The sample complexity graphs may be seen in Figure 3.

IV. Results & Discussion

The monomials of best fit are summarised in Table II. **For the task DO2**, we observe a gap in the required number of samples that grows at a roughly cubic rate

TABLE II: Monomials of best fit

$m(d)$	**DO2**	**NFP**
FCNN	$0.01d^{3.36}$	$41.68d^{1.31}$
CNN	$438.39d^{0.46}$	$23090d^{0}$

of $d^{2.91}$. This gap cannot be explained with the results established in Li. et. al. (2020) [1] since they prove either a quadratic gap for the case of all input distributions with h_2 labelling $\times$ orthogonal-equivariant learning algorithms, or a linear gap for a particular distribution $\times$ permutation-equivariant learning algorithms. Furthermore, we observe that our max-pooling CNN requires training data at rate of roughly $\sqrt{d}$, which is in contrast to the $O(1)$ result obtained for the simplified CNN with quadratic activations on the same task in Li et. al. (2020).

For the task NFP, we observe a gap that grows at a superlinear rate of $d^{1.31}$. We remark that the CNN appears to require a lot of training samples for smaller input dimensions around $d = 10^2$, for which we don't have a clear explanation. The CNN's sample complexity eventually tapers to a constant, which could be explained by the CNN potentially learning a small constant number of filters that correspond to high frequency signals such as noise, and classifying the rest as structured images.

V. Conclusion

We construct a naturalistic task based on Fourier patterns which induces a sample complexity gap that we estimate to be at least linear in the input dimension d between realistic, over-parameterised, parameter-matched FCNN and CNN models trained with Adam. We also observe a sample complexity gap that is roughly cubic in d for the difference of 2-norms task. This is stronger than the square gap predicted by Li et al. (2020) [1], despite the fact that we are in the more realistic over-parameterised regime for both models. An interesting avenue for future research could be to explore the possibility of rigorously proving the gaps we observe.

References

[1] Z. Li, Y. Zhang, and S. Arora, "Why are convolutional nets more sample-efficient than fully-connected nets?" *arXiv preprint arXiv:2010.08515*, 2020.

[2] A. Brutzkus and A. Globerson, "An optimization and generalization analysis for max-pooling networks," in *Uncertainty in Artificial Intelligence*. PMLR, 2021, pp. 1650–1660.

[3] S. Shalev-Shwartz and S. Ben-David, *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[4] A. Blumer, A. Ehrenfeucht, D. Haussler, and M. K. Warmuth, "Learnability and the vapnik-chervonenkis dimension," *Journal of the ACM (JACM)*, vol. 36, no. 4, pp. 929–965, 1989.

[5] S. Hanneke, "The optimal sample complexity of pac learning," *Journal of Machine Learning Research*, vol. 17, no. 38, pp. 1–15, 2016. [Online]. Available: http://jmlr.org/papers/v17/15-389.html

[6] N. Harvey, C. Liaw, and A. Mehrabian, "Nearly-tight vc-dimension bounds for piecewise linear neural networks," in *Conference on learning theory*. PMLR, 2017, pp. 1064–1068.