DIPLOMARBEIT

The Vapnik-Chervonenkis Dimensions of Different Neural Network Architectures

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Technische Mathematik

eingereicht von

Sebastian Bittner, BSc
Matrikelnummer 11776808

ausgeführt am Institut für Analysis und Scientific Computing
der Fakultät für Mathematik und Geoinformation
der Technischen Universität Wien

Betreuung
Betreuer: Associate Prof. Dipl.-Ing. Dr.techn. Clemens Heitzinger
Mitwirkung: Univ.-Prof. Dipl.-Ing. Dr.rer.nat. Radu Grosu

Wien, am 22.05.2023

(Unterschrift Verfasser)

(Unterschrift Betreuer)

Contents

Abstract

The Vapnik-Chervonenkis dimension, VC dimension in short, is a measure of expressivity or richness of a set of functions. In this thesis, we explore this concept in relation to different neural network architectures that use sigmoid activation functions. More specifically, we will take a look at classical multilayered feed-forward neural networks and at two NeuralODE architectures, namely Liquid Time Constant (LTC) networks and Continuous-Time Recurrent Neural Networks (CT-RNNs). In the latter two, the output of the network is computed by numerically solving an ordinary differential equation.

For these networks, we derived upper bounds on the VC dimension, depending on the number of neurons, and in case of the recurrent models (LTC and CT-RNN), discretization steps. This was done through a method involving the number of components of the zero-set of functions that are dependent on the network parameters. Here various techniques relating to topology and geometrical analysis were used. We find a very strong dependence of the VC dimension bound on the number of neurons and a sizeable dependence on the number of discretization steps. The recurrent models had a higher bound than the classical network for the same number of neurons, which is partly due to the recurrent models having more parameters than the classical network.

Kurzfassung

Die Vapnik-Chervonenkis-Dimension, kurz VC-Dimension, ist ein Maß für die Expressivität einer Menge von Funktionen. In dieser Arbeit untersuchen wir dieses Konzept in Bezug auf verschiedene neuronale Netzwerke, die die Sigmoidfunktion als Aktivierungsfunktion verwenden. Genauer gesagt betrachten wir klassische Multilayered Feed-forward Netzwerke und zwei NeuralODE-Architekturen, nämlich Liquid Time Constant(LTC) Netzwerke und Continuous-Time Recurrent Neural Networks(CT-RNNs). In den NeuralODE-Architekturen wird die Ausgabe durch numerisches Lösen einer gewöhnlichen Differentialgleichung berechnet, wovon sich auch der Name ableitet.

Für diese Netzwerkklassen leiten wir obere Schranken für die VC Dimension ab, und zwar in Abhängigkeit von der Anzahl der Neuronen, und im Falle der NeuralODE Modelle auch in Abhängigkeit von der Anzahl der Diskretisierungsschritte. Hierzu verwenden wir eine Methode, bei der die Anzahl der Komponenten der Nullmengen von Funktionen – die von den Parametern des Netzwerks abhängen – eine wesentliche Rolle spielt. Dabei finden verschiedene Methoden aus Topologie und Geometrischer Analysis Anwendung. Wir finden einen starken Zusammenhang zwischen der VC-Dimensions-Schranke und der Anzahl der Neuronen im Netzwerk. Die Abhängigkeit von der Anzahl der Diskreditierungsschritte ist auch gegeben, allerdings in geringerem Ausmaß. Die Schranke der NeuralODE-Netze ist etwas höher als die der klassischen Netze bei gleicher Anzahl an Neuronen, was zum Teil darauf zurückzuführen ist, dass die NeuralODE Netze mehr Parameter haben.

Acknowledgements

I would like to thank my supervisor Clemens Heitzinger for his continuous support and ideas as to which direction to take the thesis. I would also like to thank my co-supervisor Radu Grosu, as it was his willingness to introduce me to his work that led to me writing on this topic. Finally, I would like to thank my family, friends and colleagues, as it was only due to their support of various kind that I was able to advance and enjoy my studies and finally finish this thesis.

1 Introduction

Neural networks have been around since the 1950s and have become very relevant once computers became fast enough to handle all the required operations in a reasonable time [Ano]. Since then, numerous different artificial neural network types have emerged, and oftentimes the question arises as to which type is best suited for the task at hand. This means that there is a necessity for some sort of comparison between the types. In this thesis, we will take a theoretic approach by looking at bounds of the VC-dimension. We will be looking at classical feed-forward models, but also at two different NeuralODE architectures, namely continuous time recurrent neural networks (CT-RNNs) and Liquid Time Constant networks (LTCs). Note that we only consider networks using the sigmoid activation function (defined as $\sigma(x) := 1/(1+e^{-x})$), since our approach is limited to these.

The VC-dimension, short for Vapnik-Chervonenkis-dimension, is named after Russian mathematicians Vladimir Vapnik and Alexey Chervonenkis. These two authors introduced this concept in 1968, and a translation of their paper can be found in [VC71]. It is typically seen as a measure of richness, expressive power or flexibility of a function class. In the 1980s and 1990s, this definition was then applied to the emerging neural networks. For example, Eric Baum and David Haussler found a bound for the VC-dimension for linear threshold networks in 1989 [BH89]. A few years later, a bound was found for networks that use so-called Pfaffian activation functions [KM97]. The most relevant use-case are networks with the sigmoid activation function, for which the proof will be found in Theorem 4.1.2. This proof will then serve as a blueprint for the proofs of VC-dimension bounds for both LTC and CT-RNN networks. Most of this thesis is based on the book *Neural Network Learning: Theoretical Foundations* by Martin Anthony and Peter Bartlett [AB99], which was released in 1999 and summarizes all the results that were obtained the years before. The interest in these theoretic foundations seems to have quieted down since then, since we are not aware of any major breakthrough in the VC-dimension theory of neural networks after 1999.

While feed-forward neural networks have been around for a long time, NeuralODE architectures (including CT-RNNs and LTCs) have garnered attention in recent years. They are comprised of neurons which are much more complicated than those in feed-forward networks. The value of such a neuron changes over time based on the values of all other neurons according to a certain ordinary differential equation. These approaches can be used for processes that require a continuous output from the network, for example when predicting time series, or when training agents in a physics simulation. [HLA+20]

The outline of the thesis is as follows. In Chapter 2 we will introduce the Vapnik Chervonenkis dimension and the growth function, the two most important concepts in this thesis. We will then continue in Chapter 3 with a number of results about the VC-dimension of certain function classes. In this chapter, we will lay the groundwork for Chapter 4, where we use these insights to find bounds to the VC-dimension of neural networks. There, we will figure out the most important factors that determine the bound. These are the number of neurons, the number of parameters and in case of the NeuralODE architectures, the number of integration steps. It turns out, that the difference between the CT-RNN and LTC networks is small, and that the only important distinction between the two is the number of parameters per neuron. In Chapter 5 we will finally summarize the most important insights of the thesis.

2 The VC-Dimension and Growth Function

2.1 Definition

In this section we will introduce the notions of VC-dimension and growth function, which will be integral to the remainder of this thesis. The definition, Lemma 2.2.5 and Theorem 2.3.1 are based on [AB99].

Definition 2.1.1. *We define the* binary sign *function on* $\mathbb{R}$ *as*

$$\operatorname{bsgn}(x) := \begin{cases} 1, & \textit{if } x > 0, \\ 0, & \textit{else.} \end{cases}$$

Definition 2.1.2. *Let H be a class of functions mapping from a set X to $\{0,1\}$. We shall call a function that maps to $\{0,1\}$ a* dichotomy*. The growth function $\Pi_H \colon \mathbb{N} \to \mathbb{N}$ is defined as*

$$\Pi_H(m) := \max_{S \subseteq X, |S|=m} \left| H|_S \right|,$$

where $H|_S := \{h|_S : h \in H\}$ and $|\cdot|$ denotes the cardinality of a set.

A set $S \subseteq X$ is said to be shattered *by H, if for every dichotomy h' on S there exists $h \in H$ with $h|_S = h'$. The* Vapnik-Chervonenkis dimension *of H, or* $\operatorname{VCdim}(H)$ *is the largest number n such that there exists a set $S \subseteq X$ with cardinality n that is shattered by H. Clearly, this is the largest n such that $\Pi_H(n) = 2^n$, since there exist 2^n dichotomies on a set of cardinality n. If no such n exists, we say that* $\operatorname{VCdim}(H) = \infty$*.*

2.2 Examples

Let us look at a few examples to illustrate this idea.

Example 2.2.1. Consider

$$H := \{(x \mapsto \operatorname{bsgn}(x-a)) : a \in \mathbb{R}\},$$

the class of linear threshold functions on $\mathbb{R}$. For any $S \subseteq \mathbb{R}$, $H|_S$ then contains all functions $S \to \{0,1\}$ that are non-decreasing. Since there exist $|S|+1$ such functions, $\Pi_H(m) = m+1$. Therefore the VC-dimension of H is only 1.

Example 2.2.2. Let $X = \mathbb{R}$ and H be the set of functions of the form $x \mapsto \operatorname{bsgn}(p(x))$ for some polynomial p with degree at most d. Then H has VC-dimension $d+1$. To see this, choose some $x_1 < x_2 < \cdots < x_{d+1}$ in $\mathbb{R}$. Take $y_1, \ldots, y_d$ such that $x_i < y_i < x_{i+1}$ for all i. Given a dichotomy h on $S := \{x_1, \ldots, x_{d+1}\}$, define the polynomials

$$\begin{aligned} p_0(x) &:= 1, \\ p_{i+1}(x) &:= \begin{cases} p_i(x), & \text{if } \operatorname{bsgn}(p_i(x_{i+1})) = h(x_{i+1}), \\ p_i(x)\cdot(y_i - x), & \text{else.} \end{cases} \end{aligned}$$

Via a simple induction one can show that $\operatorname{bsgn}(p_i)$ corresponds with h on the points $x_1, \ldots, x_i$ and has degree at most $i-1$. So $\operatorname{bsgn}(p_{d+1})$ fulfills the requirements. Conversely, if H shatters a set of points $x_1, \ldots, x_n$, then there exists a polynomial p whose (binary) sign alternates at these points. If p is nonzero at all these points, then the polynomial has roots in between the points, and so $n - 1 \leq \deg p = d$. If $p(x_i) = 0$ for some i and this root is the only one in $[x_{i-1}, x_{i+1}]$, then it is a double root such that $n \leq d+1$ also holds.

Example 2.2.3. We consider the class of linear classifiers on $\mathbb{R}^2$,

$$H := \{(\mathbb{R}^2 \ni x \to \operatorname{bsgn}(y \cdot x - b)), y \in \mathbb{R}^2, b \in \mathbb{R}\}.$$

How can we determine for a function $g\colon S \to \{0,1\}$ for some finite $S \subseteq \mathbb{R}^2$, if $g \in H|_S$? If we consider the line $\{x : h(x) = 0\}$ and the two half-planes generated by that line, then h maps one of these half-planes to 0, and the other to 1 (and maps the line to 0 by our definition of bsgn). The function $1 - h$, which is also in H, maps the planes to the opposite value. So $g \in H|_S$ if and only if one can find a line that separates all the points x where $g(x) = 0$ and those where $g(x) = 1$.

The VC-dimension of this class is clearly at least 3, since one can just consider 3 points that are not in the same line. For any $g \in H|_S$, we can find a line that separates the points in the required way. Can we go further than that. What if we take S to be the verteces of a square, $\{(0,0),(0,1),(1,0),(1,1)\}$? This set cannot be shattered by H, since the points $(0,1)$ and $(1,0)$ cannot be separated from $(0,0)$ and $(1,1)$, so the function $g := ((x,y) \mapsto x + y \mod 2)$ is not in $H|_S$. So can we find other points? From geometrical intuition it seems not to be possible. Indeed, one can show that the VC-dimension is only 3. A proof can be found in [AB99, Theorem 3.1 and Page 36].

Example 2.2.4. Consider now the class of functions

$$H := \{\begin{pmatrix} x \\ y \end{pmatrix} \mapsto \operatorname{bsgn}(a_1x^2 + a_2xy + a_3y^2 + a_4x + a_5y + a_6) : a_1, \ldots, a_6 \in \mathbb{R}\}$$

of second degree polynomials in two variables. Now we can represent the function that maps $(0,0)$ and $(1,1)$ to 1 and $(0,1)$ and $(1,0)$ to 0. In fact, $\operatorname{bsgn}(4xy - 2x - 2y + 1)$ does exactly that. Hence the VC-dimension is at least 4, since one can see that all other dichotomies can also be expressed. The question whether or not one can shatter larger sets seems to be quite complicated.

From these examples it seems unlikely that a function class that looks simple can have infinite VC-dimension. But that is not the case as shown by the next lemma.

Lemma 2.2.5. *The class of functions* $F := \{(x \mapsto \operatorname{bsgn}(\sin(\theta x)) : \theta \in \mathbb{R}\}$ *from* $\mathbb{R}$ *to* $\{0,1\}$ *shatters every finite subset of* $\{2^m : m \in \mathbb{N}\}$.

Proof. We show that for $x_i := 2^{i-1}$ and a function $h\colon\ S_k \to \{0,1\}$, where $S_k := x_1, \ldots, x_k$, we can find $f \in F$ such that $f|_{S_k} = h$. We define $h' := 1 - h$ and choose c such that its binary representation is the same as the sequence $(h'(x_1), \ldots, h'(x_k), 1)$, which means

$$c := \sum_{i=1}^{k} 2^{-i} h'(x_i) + 2^{-(k+1)}.$$

Choose θ to be $2\pi c$ and observe that

$$\begin{aligned}\sin(\theta x_i) &= \sin\left(2\pi\left(\sum_{j=1}^{k} 2^{-j}h'(x_j)) + 2^{-(k+1)}\right)2^{i-1}\right)\\ &= \sin\left(\pi\left(\sum_{j=1}^{i-1}\underbrace{2^{i-j}h'(x_j)}_{\in 2\mathbb{Z}} + h'(x_i) + \sum_{j=i+1}^{k} 2^{i-j}h'(x_j) + 2^{i-k-1}\right)\right)\\ &= \sin\left(\pi h'(x_i) + \underbrace{\pi\left(\sum_{j=i+1}^{k} 2^{i-j}h'(x_j) + 2^{i-k-1}\right)}_{\in(0,\pi)}\right).\end{aligned}$$

Therefore we see that $\sin(\theta x_i) < 0$ if $h'(x_i) = 1$ and $\sin(\theta x_i) > 0$ if $h'(x_i) = 0$. This implies that $\text{bsgn}(\sin(\theta x_i)) = 1 - h'(x_i) = h(x_i)$, as required. Since h was arbitrary, F shatters S_k. Every finite subset of $\{2^m : m \in \mathbb{N}\}$ is also a subset of some S_k, and the claim follows.

□

2.3 The Relation between Growth Function and VC-Dimension

The next theorem tightly relates the growth function to the VC dimension.

Theorem 2.3.1. *For a function class with* $\text{VCdim}(H) = d$*, the inequality*

$$\Pi_H(m) \le \sum_{i=0}^{d}\binom{m}{i} < \left(\frac{em}{d}\right)^d$$

holds for $m > d$*.*

Proof. Fix a set $S = \{x_1, \dots, x_m\} \subseteq X$. Instead of the restrictions of the functions to the set S, we consider the set

$$\mathcal{F} := \{\{x \in S : f(x) = 1\} : f \in H\}.$$

There is clearly a one-to-one correspondence between these and $H|_S$, since the elements of the latter are just the characteristic functions to the sets in $\mathcal{F}$. So $\left|H|_S\right| = |\mathcal{F}|$. We will now transform $\mathcal{F}$ into a family of sets that has the same cardinality.

Define a function T_x for $x \in S$ as

$$T_x := \left\{\begin{array}{ccc} \mathcal{P}(\mathcal{P}(S)) & \to & \mathcal{P}(\mathcal{P}(S)),\\ \mathcal{G} & \mapsto & \{A \setminus \{x\} : A \in \mathcal{G}\} \cup \{A \in \mathcal{G} : A \setminus \{x\} \in \mathcal{G}\}.\end{array}\right.$$

The effect of T_x is that from each set A in $\mathcal{G}$, we remove x in case the set $A \setminus \{x\}$ is not in $\mathcal{G}$ already. T_x preserves cardinality, since

$$f(A) := \left\{\begin{array}{cc} A, & \text{if } A \setminus \{x\} \in \mathcal{G},\\ A \setminus \{x\}, & \text{else},\end{array}\right.$$

is a bijection from $\mathcal{G}$ to $T_x(\mathcal{G})$.

To see this, first notice that if $W := f(A_1) = f(A_2)$ and $A_1 \neq A_2$, one of these sets is W and the other $W \cup \{x\}$. W.l.o.g. we have $W = A_1 \in \mathcal{G}$. But then $A_2 \setminus \{x\} = W \in \mathcal{G}$ and therefore $f(A_2) = A_2 \neq A_1 = f(A_1)$, a contradiction. For the surjectivity, let $A \in T_x(\mathcal{G})$. Either $A, A \setminus \{x\} \in \mathcal{G}$, then $f(A) = A$, or $A \notin \mathcal{G}$ and $A = B \setminus \{x\}$ for some $B \in \mathcal{G}$, in which case $f(B) = A$.

We say that $\mathcal{H}$ is closed under the subtraction of $x \in S$, if

$$A \in \mathcal{H} \Rightarrow A \setminus \{x\} \in \mathcal{H}.$$

Clearly, $T_x(\mathcal{G})$ has this property. We now show that $T_x(\mathcal{G})$ is closed under the subtraction of any $y \in S$, if $\mathcal{G}$ already is. To this end, suppose $B \in T_x(\mathcal{G})$. In case $B = A \setminus \{x\}$ for some $A \in \mathcal{G}$, then $B \setminus \{y\} = A \setminus \{x, y\} \in T_x(\mathcal{G})$, since $A \setminus \{y\} \in \mathcal{G}$. In case that B and $B \setminus \{x\}$ are both in $\mathcal{G}$, so are $B \setminus \{y\}$ and $B \setminus \{x, y\}$. Therefore $B \setminus \{y\}$ is also in $T_x(\mathcal{G})$.

Now define $\mathcal{F}^* := T_{x_1} \circ T_{x_2} \circ \cdots \circ T_{x_m}(\mathcal{F})$. By what we have just proven, $\mathcal{F}^*$ is closed under subtraction of all $x \in S$. In particular, if $B \subseteq A \in \mathcal{F}^*$, then already $B \in \mathcal{F}^*$.

The next step is to prove that any set shattered by $\mathcal{F}^*$ is also shattered by $\mathcal{F}$. Here, $R \subseteq S$ is said to be shattered by $\mathcal{G}$ if $\mathcal{G} \cap R := \{A \cap R : A \in \mathcal{G}\} = \mathcal{P}(R)$. If $x \in S$ and $R \subseteq R$, suppose that $T_x(\mathcal{F})$ shatters R. If $x \notin R$, then of course $T_x(\mathcal{F}) \cap R = \mathcal{F} \cap R = \mathcal{P}(R)$. Suppose $x \in R$. For any $A \subseteq R$ with $x \notin A$, we have $A \cup \{x\} = B \cap R$ for some $B \in T_x(\mathcal{F})$(and $x \in B$. By the definition of T_x, B and $B \setminus \{x\}$ are in $\mathcal{F}$, and $A = B \setminus \{x\} \cap R$. Therefore both A and $A \cup \{x\}$ are in $\mathcal{F} \cap R$, which implies that $\mathcal{F}$ shatters F.

Since $\mathcal{F}$ has VC-dimension d, $\mathcal{F}^*$ can shatter sets with cardinality no more than d. But $\mathcal{F}^*$ shatters every set that it contains, since every subset of a set in $\mathcal{F}^*$ is again in $\mathcal{F}^*$. So every set in $\mathcal{F}^*$ has at most cardinality d, which implies

$$|\mathcal{F}^*| \leq \sum_{i=0}^{d} \binom{m}{i}. \tag{1}$$

By the Binomial Theorem and Euler's inequality ($(1 + a/n)^n < e^a$),

$$\sum_{i=0}^{d} \binom{m}{i} \leq \left(\frac{m}{d}\right)^d \sum_{i=0}^{d} \binom{m}{i} \left(\frac{d}{m}\right)^d < \left(\frac{me}{d}\right)^d, \tag{2}$$

which finishes the proof.

Example 2.3.2. We know that the linear classifiers from Example 2.2.3 have VC-dimension 3. Theorem 2.3.1 now gives the bound

$$\Pi_H(m) \leq \sum_{i=0}^{3} \binom{m}{i} \leq \frac{m^3}{6} + \frac{m^2}{2} + m + 1$$

for the growth function. □

3 Bounding the VC-Dimension using Geometric Techniques

3.1 The Relationship between Solution Set Components and the Growth Function

In this section, we will find a way to bound the growth functions of a certain type of parameterized function class (so-called k-combinations, see Definition 3.1.1) by the number of components of intersections of zero sets of corresponding functions in parameter space. This will be the main result of this section, Theorem 3.1.10, whereby all other results will be leading up to that theorem. We follow closely [AB99], and all proofs are based on this book, unless stated otherwise.

Definition 3.1.1. *Let H be a class of $\{0,1\}$-valued functions defined on a set X, and F a class of real-valued functions defined on $\mathbb{R}^d \times X$. We call H a k-combination of* $\operatorname{bsgn}(F)$ *if there is a function $g\colon \{0,1\}^k \to \{0,1\}$ and functions $f_1, \dots, f_k$ in F such that for any $h \in H$ there is a parameter vector $a_h \in \mathbb{R}^d$ and*

$$h(x) = g(\operatorname{bsgn}(f_1(a_h, x)), \dots, \operatorname{sgn}(f_k(a_h, x))) \tag{3}$$

for all $x \in X$.

Example 3.1.2. Let F be the class of linear functions from $\mathbb{R} \times \mathbb{R}$ to $\mathbb{R}$. The class of threshold functions

$$H := \{(x \mapsto \operatorname{bsgn}(x - a)) : a \in \mathbb{R}\}$$

is a 1-combination of $\operatorname{bsgn}(F)$ *by the choices $f_1(a, x) := x - a$ and $g(b) = b$.*

Definition 3.1.3. *A finite set $\mathcal{F}$ of differentiable functions mapping from $\mathbb{R}^d$ to $\mathbb{R}$ is said to have* regular zero-set intersections *if for all nonempty subsets $\{f_1, \dots, f_k\} \subseteq \mathcal{F}$, the Jacobian of $(f_1, \dots, f_k)\colon \mathbb{R}^d \to \mathbb{R}^k$ has rank k at every point of the solution set*

$$\bigcap_{i=1}^{k} \{a \in \mathbb{R}^d : f_i(a) = 0\}.$$

Note that for every subset of $\mathcal{F}$ that contains more than d functions, the solution set of these functions must be empty since the rank condition cannot be satisfied.

Before we state the main results, we will first have a look at the following lemma. It posits that having regular zero-set intersection is the usual case, in the sense that given any functions, almost all translations of these functions lead to a set with regular zero-set intersections.

Lemma 3.1.4. *Given a finite sequence $f_1, \dots, f_k \in C^d(\mathbb{R}^d; \mathbb{R})$, define*

$$S := \{c \in \mathbb{R}^k : \\ \{f_i - c_i : i = 1, \dots, k\} \text{ does not have regular zero-set intersections}\}.$$

Then $\lambda(S) = 0$, where λ denotes the k-dimensional Lesbegue measure.

Proof. Consider a sequence of distinct indizes $A := (i_1, \dots, i_\ell)$ and let $f_A := (f_{i_1}, \dots, f_{i_\ell})$. By Sard's Theorem (see [Ste64, Theorem 3.1])

$$S_A := f_A(x \in \mathbb{R}^d : df_A(x) < \ell\})$$

has (ℓ-dimensional) measure 0. Therefore the complement of $T_A := \{c \in \mathbb{R}^k : (c_{i_1}, \dots, c_{i_\ell}) \notin S_A\}$ also has (k-dimensional) measure 0. Now define

$$T := \bigcap_{A \in I} T_A,$$

where I is the set of all eligible index sequences.

Let $c \in S$ be arbitrary. Since $\{f_1 - c_1, \dots, f_k - c_k\}$ does not have regular zero-set intersections, there exists m and an index sequence $(i_1, \dots, i_m) \in I$ such that the Jacobian of $(f_{i_1} - c_{i_1}, \dots, f_{i_m} -$

$c_{i_m})$ at some point x of the solution set does not have rank m. Since $f_{i_j}(x) = c_{i_j}$ for $j = 1, \dots, m$, this implies $(c_{i_1}, \dots, c_{i_\ell})^T \in S_{\{i_1,\dots,i_\ell\}}$ and so $c \notin T_{\{i_1,\dots,i_\ell\}} \supseteq T$. Since $c \in S$ was arbitrary, we have $S \subseteq T^c$. Since T^c is the finite union of the zero-sets T_A^c it is again a zero-set and consequently, so is S. □

The following lemma will now make a connection between the growth function of a k-combination of $\operatorname{bsgn}(F)$, when F is a function class, and the number of connected components of the complements of the zero-sets of these functions.

Lemma 3.1.5. *Let F be a class of functions $\mathbb{R}^d \times X \to \mathbb{R}$ that is closed under the addition of constants. Suppose that $f(.,x)$ is d times continuously differentiable for every $f \in F$. Let H be a k-combination of $\operatorname{bsgn}(F)$. Then there exist $f_{i,j} \in F$ $(i = 1, \dots, k,\ j = 1, \dots, m)$ and $x_1, ..., x_m \in X$ such that the number of connected components of*

$$\mathbb{R}^d \setminus \bigcup_{i=1}^{k} \bigcup_{j=1}^{m} \{a \in \mathbb{R}^d : f_{i,j}(a, x_j) = 0\}$$

is at least $\Pi_H(m)$ and the set

$$\{(a \mapsto f_{i,j}(a, x_j)) : i = 1, \dots, k, j = 1, \dots, m\}$$

has regular zero-set intersections.

Proof. Let $\tilde{f}_1, \dots, \tilde{f}_k \in F$ and $g\colon \{0,1\}^k \to \{0,1\}$ such that for every $h \in H$ we can find $a_h \in \mathbb{R}^d$ that satisfies

$$h(x) = g(\operatorname{bsgn}(\tilde{f}_1(a_h, x)), \dots, \operatorname{bsgn}(\tilde{f}_k(a_h, x))). \tag{4}$$

Setting $n := \Pi_H(m)$, we can find $x_1, ..., x_m \in X$ and $h_1, \dots, h_n \in H$, such that no two distinct functions correspond on every x_i. For h_j we can now find a corresponding coefficient $a_j := a_{h_j}$ according to (4). Choose ε strictly between 0 and

$$\min\{f_i(a_\ell, x_j) : f_i(a_\ell, x_j) > 0,\ i = 1, \dots, k,\ j = 1, \dots, m,\ \ell = 1, \dots, n\}.$$

If this set is empty, choose any $\varepsilon > 0$. Define $I := \{1, \dots, k\} \times \{1, \dots, m\}$ and consider the set

$$\begin{aligned} C := (0, \varepsilon)^{km} \cap \{(c_{i,j})_{(i,j) \in I} \in \mathbb{R}^{km} : \{f_i(., x_j) - c_{i,j} : (i,j) \in I\} \\ \text{has regular zero-set intersections}\}. \end{aligned}$$

By Lemma 3.1.4 C is the intersection of a set with positive measure with a set whose complement has measure zero, therefore $\lambda(C) > 0$. In particular, $C \neq \emptyset$. Now choose any $c_{i,j} \in C$ and define $f_{i,j} := \tilde{f}_i - c_{i,j}$. Every $f_{i,j}$ is in F since this class is closed under the addition of constants. According to our choice of ε, we have that

$$\operatorname{bsgn}(\tilde{f}_i(a_\ell, x_j)) = 1 \ \Rightarrow\ f_{i,j}(a_\ell, x_j) > 0, \tag{5}$$

$$\operatorname{bsgn}(\tilde{f}_i(a_\ell, x_j)) = 0 \ \Rightarrow\ f_{i,j}(a_\ell, x_j) < 0 \tag{6}$$

for all i, j and ℓ.

Consider now the set

$$K := \mathbb{R}^d \setminus \bigcup_{i=1}^{k} \bigcup_{j=1}^{m} \{a \in \mathbb{R}^d : f_{i,j}(a, x_j) = 0\}.$$

For $\ell \neq \ell'$ and j such that $h_\ell(x_j) \neq h_{\ell'}(x_j)$ we can find i such that

$$\operatorname{bsgn}(\tilde{f}_i(a_\ell, x_j)) \neq \operatorname{bsgn}(\tilde{f}_i(a_{\ell'}, x_j)),$$

w.l.o.g. we have $\operatorname{bsgn}(\tilde{f}_i(a_\ell, x_j)) = 0$, $\operatorname{bsgn}(\tilde{f}_i(a_{\ell'}, x_j)) = 1$. By (5) and (6) this implies

$$f_{i,j}(a_\ell, x_j) < 0 < f_{i,j}(a_\ell, x_j)$$

and $a_\ell, a'_\ell \in K$. In addition, there cannot be a continuous path in K that connects a_ℓ and $a_{\ell'}$. In fact, for any such path $\gamma\colon [b,c] \to \mathbb{R}^d$ connecting a_ℓ and a'_ℓ, there must exist $t \in [b,c]$ such that $f_{i,j}(\gamma(t), x_j) = 0$, so $\gamma(t)$ is not in K. Since the considered set is open, the path-components and the connected components coincide. (See [Kal14, Lemma 11.3.3 and Proposition 11.3.6.].) Hence a_ℓ and a'_ℓ are not in the same connected component. In conclusion, $a_1, \ldots, a_n$ all lie in different components of K, therefore $CC(K) \geq n$. □

Lemma 3.1.5 is already a very remarkable result. However, we would like to have a bound that is dependent on the number of components of intersections of the zero-sets, not the complement.

Lemma 3.1.6. *Let $f_1, \ldots, f_k$ be differentiable functions mapping from $\mathbb{R}^d$ to $\mathbb{R}$ with regular zero-set intersections. For $i = 1, \ldots, k$ define $Z_i := f_i^{-1}(\{0\})$. Then the inequality*

$$CC\left(\mathbb{R}^d \setminus \bigcup_{i=1}^{k} Z_i\right) \leq \sum_{S \subseteq \{1,\ldots,k\}} CC\left(\bigcap_{i \in S} Z_i\right)$$

holds.

The prove of this lemma is quite long, so it will be split up in multiple sub-results. The proof of the first such subresult, Lemma 3.1.7, is based on [Kal18, Satz 13.4.4] and [War68, Lemma 1.1.].

Lemma 3.1.7. *Define functions $f_1, \ldots, f_k$ as in Lemma 3.1.6 and let $O \subseteq \mathbb{R}^d$ open. Consider the sets $Z_i := f_i^{-1}(\{0\})$ for $i = 2, \ldots, k$. Let C be a connected component of $M := \bigcap_{i=2}^{k} Z_i \cap O$ and D a connected component of $N := f_1^{-1}(\{0\})$. Then $C \setminus D$ has at most two connected components.*

Proof. Let $y \in D^* := C \cap D$. We will show that C and D^* are both manifolds, and that the latter is embedded in the former in a special way. This part of the proof closely follows the proof of [Kal18, Satz 13.4.4]. Since the involved functions have regular zero-set-intersections, the Jacobian of $(f_1, \ldots, f_k)$ has rank k, in other words, there exist k coordinates $x_1, \ldots x_k$ such that

$$\left(\frac{\partial f_\ell(y)}{\partial x_{i_j}}\right)_{j,\ell=1}^{k} \tag{7}$$

is a regular matrix. Letting $p\colon \mathbb{R}^d \to \mathbb{R}^{d-k}$ be the projection onto the remaining coordinates, define the map

$$\varphi(x) := \begin{pmatrix} p(x) \\ f_1(x) \\ \vdots \\ f_k(x) \end{pmatrix},$$

which will be a chart that will make D^* a $(d-k)$-dimensional manifold. This function is C^1, where

$$\frac{\partial \varphi(x)}{\partial x_i} = \begin{pmatrix} 0 \\ \frac{\partial f_1(y)}{\partial x_i} \\ \vdots \\ \frac{\partial f_k(y)}{\partial x_i} \end{pmatrix}$$

for $i \in \{i_1, \dots, i_k\}$ and

$$\frac{\partial \varphi(x)}{\partial x_i} = \begin{pmatrix} e_i \\ \frac{\partial f_1(y)}{\partial x_i} \\ \vdots \\ \frac{\partial f_k(y)}{\partial x_i} \end{pmatrix}$$

for the remaining coordinates. Here e_i denotes the i'th unit vector.

Therefore the Jacobian of φ has the form of a block-diagonal matrix (after a suitable column permutation) that has the identity matrix and the matrix in (7) in the diagonal, and is therefore regular. By the inverse mapping theorem, [Kal18, Korollar 13.2.1], there exist open $U_y \subseteq \mathbb{R}^d$ and $V_y \subseteq D$ such that $\varphi|_{U_y}$ is a diffeomorphism onto V_y and $y \in U_y$. Since the restriction of a diffeomorphism is again one, we can make these sets smaller such that U_y is contained in the open set $O \setminus (\overline{M \setminus C} \cup \overline{N \setminus D})$ and such that V_y is a ball. Notice that this is possible since $\overline{M \setminus C} \cap C = \emptyset$, $\overline{N \setminus D} \cap D = \emptyset$ and therefore $y \in O \setminus (\overline{M \setminus C} \cup \overline{N \setminus D})$. This restriction means that U_y is contained in O and is disjoint from other connected components of C and D. Consequently we have for $x \in U_y$ that

$$\begin{aligned} x \in D^* \iff x \in M \cap N \iff & (f_1(x), \dots, f_k(x)) = 0 \\ \iff & \varphi(x) \in \mathbb{R}^m \times \{0\} \cong \mathbb{R}^m \end{aligned}$$

and

$$\begin{aligned} x \in C \iff x \in M \iff & (f_2(x), \dots, f_k(x)) = 0 \\ \iff & \varphi(x) \in \mathbb{R}^{m+1} \times \{0\} \cong \mathbb{R}^{m+1}. \end{aligned}$$

Let $h \colon U_y \cap C \to \tilde{V}_y$ be the homeomorphism $p' \circ \varphi$, where $p' \colon \mathbb{R}^d \to \mathbb{R}^{m+1}$ is the projection onto the first $m+1$ coordinates and $\tilde{V}_y := p'(V_y)$. We can now split this ball according to the sign of the first coordinate into

$$\begin{aligned} \tilde{V}_y^0 &:= \{x \in \tilde{V}_y : x_1 = 0\}, \\ \tilde{V}_y^+ &:= \{x \in \tilde{V}_y : x_1 > 0\}, \\ \tilde{V}_y^- &:= \{x \in \tilde{V}_y : x_1 < 0\}. \end{aligned}$$

By the definition of φ, the set $D^* \cap U_y$ is now mapped by h onto $\tilde{V}_y^0$, and clearly $\tilde{V}_y \setminus \tilde{V}_y^0$ has two connected components, namely $\tilde{V}_y^+$ and $\tilde{V}_y^-$.

Now $(C \setminus D) \cap U_y$ (which is the same as $(C \setminus D^*) \cap U_y$) is mapped to the set $\tilde{V}_y \setminus \tilde{V}_y^0$, and since h is a homeomorphism, $(C \setminus D) \cap U_y$ also has two connected components, namely $h^{-1}(\tilde{V_y}^{+})$

and $h^{-1}(\tilde{V_y}^{-})$. Now

$$D^* \cap U_y \subseteq h^{-1}(\overline{\tilde{V_y}^{\pm}}) = \overline{h^{-1}(\tilde{V_y}^{\pm})}. \tag{8}$$

For each component C_i of the set $C \setminus D$, consider $E_i := D^* \cap \overline{C_i}$, which is closed in the trace topology of D^*. We can write this set as

$$E_i = \bigcup_{y \in E_i} D^* \cap U_y. \tag{9}$$

To see this, first notice that, suppose $C_i \cap h^{-1}(\tilde{V}_y^+) = \emptyset = C_i \cap h^{-1}(\tilde{V}_y^-)$ for some $y \in E_i$, since $C_i \cap h^{-1}(\tilde{V}_y^0) \subseteq C_i \cap D^* = \emptyset$ we have $C_i \cap h^{-1}(\tilde{V}_y) = \emptyset$ and so $y \notin \overline{C_i} \supseteq E_i$, a contradiction. Without loss of generality, $C_i \cap h^{-1}(\tilde{V}_y^+) \neq \emptyset$. Since $h^{-1}(\tilde{V}_y^+)$ is a connected subset of $C \setminus D$, it must lie entirely in one connected component of $C \setminus D$, so $h^{-1}(\tilde{V}_y^+) \subseteq C_i$. According to (8) this implies

$$D^* \cap U_y \subseteq D^* \cap \overline{h^{-1}(\tilde{V}_y^+)} \subseteq D^* \cap \overline{C_i} = E_i$$

for every $y \in E_i$. Since clearly $E_i \subseteq \bigcup_{y \in E_i} D^* \cap U_y$, this proves (9). In particular, E_i is open in the trace topology of D^*. Being both an open and closed as a subset of the connected set D^*, either $E_i = \emptyset$ or $E_i = D^*$. Suppose $E_i = D^* \cap \overline{C_i}$ is empty, that means C_i is open (connected components of open sets are open) and closed in C (since $\overline{C_i} \cap C = \overline{C_i} \cap (C \setminus D) = C_i$), therefore $C_i = C$ which is a contradiction since $C_i \subseteq C \setminus D \subsetneq C$ ($C \cap D$ is nonempty.) Hence $E_i = D^*$. In particular, for any $y \in D^*$, we have $y \in \overline{C_i}$, which implies $U_y \cap (C \setminus D) \cap C_i \neq \emptyset$, for every component C_i of $C \setminus D$. This means that $(C \setminus D)$ does not have more connected components than $U_y \cap (C \setminus D)$, which has two. This concludes the proof. □

Lemma 3.1.8. *Let $f_1, \dots, f_k$ and $Z_1, \dots, Z_k$ as in Lemma 3.1.6. Let $I \subsetneq \{1, \dots, k\}$ with $\ell \in I^c$ and $M := \bigcap_{i \in I} Z_i$. Then the inequality*

$$CC\left(M \setminus \bigcup_{j \in I^c} Z_j\right) \leq CC\left(M \setminus \bigcup_{j \in I^c, j \neq \ell} Z_j\right) + CC\left(M \cap Z_\ell \setminus \bigcup_{j \in I^c, j \neq \ell} Z_j\right)$$

holds.

Proof. Let $S := M \setminus \bigcup_{j \in I^c, j \neq \ell} Z_j$ and let $C_1, \dots, C_N$ be the connected components of S. (If S has infinitely many components, the claim is trivial.) For arbitrary $j \in \{1, \dots, N\}$ consider the set C_j. Define A_1 to be a connected component of $C_j \cap Z_\ell$. Since the set $\bigcup_{j \in I^c, j \neq \ell} Z_j$ is open, the set S fulfills the requirement for M in Lemma 3.1.7, so $C_j \setminus A_1$ has no more than two components. Let A_2 be a second component of $C_j \cap Z_\ell$ (in case it exists). Since A_2 is disjoint from A_1, it must lie entirely in one connected component of $C_j \setminus A_1$. Call these components D_1 and D_2, in such a way that $D_1 \cap A_2 = \emptyset$. (Define $D_1 := \emptyset$ in case there is only one component.) Again by Lemma 3.1.7, $CC(D_2 \setminus A_2) \leq 2$, and so

$$\begin{aligned} CC(C_j \setminus (A_1 \cup A_2)) = CC((C_j \setminus A_1) \setminus A_2) = \\ CC(D_1 \setminus A_2 \cup D_2 \setminus A_2) = CC(D_1 \cup D_2 \setminus A_2) \\ \leq 1 + CC(D_2 \setminus A_2) \leq 3 \end{aligned}$$

since $D_1 \cup D_2 = C_j \setminus A_1$. We can proceed inductively in the same way to obtain

$$CC(C_j \setminus Z_\ell) \leq CC(C_j \cap Z_\ell) + 1.$$

Therefore we can conclude the proof by calculating

$$\begin{aligned} CC\left(M \setminus \bigcup_{j \in I^c} Z_j\right) &= CC(S \setminus Z_\ell) = CC(\bigcup_{j=1}^{N} C_j \setminus Z_\ell) = \sum_{j=1}^{N} CC(C_j \setminus Z_\ell) \\ &\leq \sum_{j=1}^{N} CC(C_j \cap Z_\ell) + 1 = CC\left(C_j \cap Z_\ell\right) + N = CC(S \cap Z_\ell) + CC(S). \end{aligned}$$

(Here we use that $CC(\bigcup_{i=1}^{m} A_i) = \sum_{i=1}^{m} CC(A_i)$ for “seperated” sets A_i.) □

We can now prove Lemma 3.1.6.

Proof of Lemma 3.1.6. We prove by induction in b the following statement. Suppose $i \geq b$. Let $\{f_1, \dots, f_i\}$ and $\{Z_1, \dots, Z_i\}$ with properties as in the lemma. Let $I \subseteq \{1, \dots, i\}$ with $|I| = i - b$. Define $M_I := \bigcap_{j \in I} Z_j$. Then

$$CC(M_I \setminus \bigcup_{j \in I^c} Z_j) \leq \sum_{S \subseteq I^c} CC\left(M_I \cap \bigcap_{j \in S} Z_j\right),$$

where $M_I \cap \bigcap_{j \in \emptyset} Z_j := M_I$.

For $b = 0$, the statement reads $CC(M_I) \leq CC(M_I)$, which is clearly true.

Suppose the statement holds for some $b \in \mathbb{N} \cup \{0\}$. Consider $i \geq b+1$, $I \subseteq \{1, \dots, i\}$ with $|I| = i - b - 1$ and f_j's and Z_j's as above. Choosing some $\ell \in I^c$, we use Lemma 3.1.8 to obtain

$$CC(M_I \setminus \bigcup_{j \in I^c} Z_j) \leq CC(M_I \setminus \bigcup_{j \in I^c \setminus \{\ell\}} Z_j) + CC(M_I \cap Z_\ell \setminus \bigcup_{j \in I^c \setminus \{\ell\}} Z_j) \tag{10}$$

$$\leq \sum_{S \subseteq I^c \setminus \{\ell\}} CC\left(M_I \cap \bigcap_{j \in S} Z_j\right) + \sum_{S \subseteq I^c \setminus \{\ell\}} CC\left(M_I \cap Z_\ell \cap \bigcap_{j \in S} Z_j\right). \tag{11}$$

In the last inequality, note that we applied the induction hypothesis to $\{f_1, \dots, f_i\} \setminus \{f_\ell\}$ and I for the first expression. In the second expression we applied it to $\{f_1, \dots, f_i\}$ and $\tilde{I} := I \cup \{\ell\}$ and made use of the fact that $M_{\tilde{I}} = M_I \cap Z_\ell$. Note that in both cases the condition $|I| = i - b$ is fulfilled.

To continue the calculation in (10), we notice that the intersections in the first expression contain indizes from subsets of S without ℓ, whereas the second expressions contains all the subsets that include ℓ. This implies

$$CC(M_I \setminus \bigcup_{j \in I^c} Z_j) \leq \sum_{S \subseteq I^c} CC\left(M_I \cap \bigcap_{j \in S} Z_j\right),$$

which proves the induction. Now taking $k = i$, we obtain the required statement.

□

Definition 3.1.9. *Let G be a set of functions in $C^1(\mathbb{R}^d;\mathbb{R})$. We say that G has solution set components bound B if for any $1 \leq k \leq d$ and any $\{f_1,\dots,f_k\} \subseteq G$ that has regular zero-set intersections, we have*

$$CC\left(\bigcap_{i=1}^{k} f_i^{-1}(\{0\})\right) \leq B.$$

A set $\tilde{G}$ of functions $\mathbb{R}^d \times X \to \mathbb{R}$ (for some set X) with $f(.,x) \in C^1(\mathbb{R}^d;\mathbb{R})$ for every $f \in \tilde{G}$ and $x \in X$ has solution set component bound B with respect to the first d variables, if

$$\{f(.,x) : f \in \tilde{G}, x \in X\}$$

has solution set components bound B.

We are now ready to prove the final result, which ties it all together. We will now be able to bound the growth function of a function class using the solution set components bound from Definition 3.1.9.

Theorem 3.1.10. *Let F be a class of functions $\mathbb{R}^d \times X \to \mathbb{R}$ that is closed under the addition of constants. Suppose that $f(.,x) \in C^d(\mathbb{R}^d;\mathbb{R})$ for every $f \in F$. In addition, let H be a k-combination of* $\operatorname{bsgn}(F)$*, therefore consisting of functions $X \to \{0,1\}$. If F has solution set components bound B with respect to the first d variables, then*

$$\Pi_H(m) \leq B \sum_{i=0}^{d} \binom{mk}{i}.$$

Proof. By Lemma 3.1.5 we can find $f_{i,j} \in F$, $i = 1,\dots,k$, $j = 1,\dots,m$ and $x_1,\dots,x_m \in X$ such that

$$\Pi_H(m) \leq CC\left(\mathbb{R}^d \setminus \bigcup_{i=1}^{k}\bigcup_{j=1}^{m}\{a \in \mathbb{R}^d : f_{i,j}(a,x_j) = 0\}\right)$$

holds and $\{f_{i,j}(.,x_j), i = 1,\dots,k,\ j = 1,\dots,m\}$ has regular zero-set intersections. Applying Lemma 3.1.6 to these functions we obtain

$$\begin{aligned}\Pi_H(m) &\leq CC\left(\mathbb{R}^d \setminus \bigcup_{i=1}^{k}\bigcup_{j=1}^{m}\{a \in \mathbb{R}^d : f_{i,j}(a,x_j) = 0\}\right)\\ &\leq \sum_{S\subseteq\{1,\dots,k\}\times\{1,\dots,m\}} CC\left(\bigcap_{(i,j)\in S}\{a \in \mathbb{R}^d : f_{i,j}(a,x_j) = 0\}\right)\\ &\leq \sum_{S\subseteq\{1,\dots,k\}\times\{1,\dots,m\},|S|\leq d} B = B\sum_{i=0}^{d}\binom{mk}{i}.\end{aligned}$$

Note that this holds since for $|S| > d$ the intersection of the zero sets must be empty. □

Remark 3.1.11. In the setting of Theorem 3.1.10, if F is not closed under addition of constants, one can consider $\tilde{F} := \{f + c : f \in F, c \in \mathbb{R}\}$. This class clearly is closed under the addition of constants, and H is also a k-combination of $\tilde{F}$. Therefore Theorem 3.1.10 can be used to determine a bound for the growth function of H using the solution set components bound of $\tilde{F}$.

Example 3.1.12. Consider the linear threshold functions from Example 3.1.2. They are a 1-combination of the linear functionals on $R \times R$. We have to determine a solution set components bound of $\{(x \mapsto ax + by + c) : a, b, c, y \in \mathbb{R}\}$, which is the set of affine maps on $\mathbb{R}$. The solution set of one such function is at most 1, so that is also a solution set components bound. Applying Theorem 3.1.10 gives a growth function bound of $\binom{m}{0} + \binom{m}{1} = m+1$. As we have seen in Example 2.2.1, this bound is tight.

3.2 Solution Set Components of Circuits

Since we now have a way to bound the growth function, it is now the goal to find solution set components bounds for the function classes we are interested in. We follow again [AB99] to find a solution set components bound to function classes described by circuits (see Lemma 3.2.6). We will see later that neural networks with sigmoid activation functions can be viewed as such a circuit.

First, we introduce the quite technical concept of intermediate variables.

Definition 3.2.1. *Let G be a set of functions in $C^1(\mathbb{R}^d; \mathbb{R})$ and $\tilde{G}$ a set of functions in $C^1(\mathbb{R}^{d(n+1)})$. We say that $\tilde{G}$ computes G with n intermediate variables if, for any $1 \leq k \leq d$ and $\{f_1, \dots, f_k\} \subseteq G$, there is a set*

$$\{\tilde{f}_1, g_{1,1}, \dots, g_{1,n}, \dots, \tilde{f}_k, g_{k,1}, \dots, g_{k,n}\} \subseteq \tilde{G}$$

that satisfies the following conditions.

1. *For every $i = 1, \dots, k$ there are functions $\phi_{i,1}, \dots, \phi_{i,k}$ and open sets $O_{i,1}, \dots, O_{i,k} \subseteq \mathbb{R}^{d(n+1)}$ such that $\phi_{i,j} \in C^1(O_{i,j}; \mathbb{R})$. These functions can be written as*

$$\phi_{i,1}(a, b) = \phi_{i,1}(a),$$
$$\phi_{i,j}(a, b) = \phi_{i,j}(a, b_{i,1}, \dots, b_{i,j-1}), \qquad 2 \leq j \leq n,$$

   where $a \in \mathbb{R}^d$ and $b = (b_{i,j})_{1 \leq i \leq k, 1 \leq j \leq n} \in \mathbb{R}^{dn}$ and $(a, b) \in O_{i,j}$. The function $\phi_{i,j}$ can be thought of as computing the intermediate variables, as it only depends on the input and the previously computed variables. Note that these $\phi_{i,j}$ do not have to belong to the class $\tilde{G}$.

2. *For $i = 1, \dots, k$ and $j = 1, \dots, n$ the function $g_{i,j}$ can be written as*

$$g_{i,j}(a, b) = g_{i,j}(a, b_{i,1}, \dots b_{i,j}).$$

3. *Let $i = 1, \dots, k$ and $\ell = 1, \dots, n$ and $a \in \mathbb{R}^d$, $b \in \mathbb{R}^{dn}$ as before. If $(a, b) \in O_{i,j}$ and $b_{i,j} = \phi_{i,j}(a, b)$ for $j = 1, \dots, \ell - 1$, then $(a, b) \in O_{i,\ell}$ and*

$$g_{i,\ell}(a, b) = 0 \Longleftrightarrow b_{i,\ell} = \phi_{i,\ell}(a, b)$$

   and

$$\frac{\partial g_{i,\ell}}{\partial b_{i,\ell}}(a, \phi_{i,1}(a, b), \dots, \phi_{i,\ell}(a, b)) \neq 0.$$

   This means that the functions $g_{i,\ell}$ implicitly defines the function $\phi_{i,\ell}$.

4. *For $a \in \mathbb{R}^d$ and $b \in \mathbb{R}^{dn}$, if $b_{i,j} = \phi_{i,j}(a, b)$ for $1 \leq i \leq k$ and $1 \leq j \leq n$, then*

$$f_i(a) = \tilde{f}_i(a, b)$$

   for $1 \leq i \leq k$.

Example 3.2.2. Let $\tilde{G}$ be the class of linear functionals on $\mathbb{R}^{2d}$. Let us now try to find out which functions can be computed with one intermediate variable. Suppose we have $1 \leq k \leq d$ and functions $f_1, \ldots, f_k \in G \subseteq C^1(\mathbb{R}^d; \mathbb{R})$, whereby $\tilde{G}$ computes G with one intermediate variable. This implies the existence of functions $g_{i,1}$ and $\phi_{i,1}$ for each $1 \leq i \leq k$ that satisfy the requirements of Definition 3.2.1. Then $g_{i,1}(a, b_{i,1}) = v \cdot a + w b_{i,1}$ for some $v \in \mathbb{R}^d, w \in \mathbb{R}$, and therefore

$$\begin{aligned} \phi_{i,1}(a) = b_{i,1} &\iff g_{i,1}(a, b_{i,1}) = 0 \\ &\iff (w \neq 0 \wedge b_{i,1} = \frac{-v \cdot a}{w}) \vee (w = 0 \wedge v \cdot a = 0). \end{aligned}$$

In the first case, it must hold that $\phi_{i,1}(a, b) = -\frac{v \cdot a}{w}$, so $\phi_{i,1}$ is again linear. In the second case we cannot find a corresponding function $\phi_{i,1}$. Therefore $f_i = \tilde{f}_i(a, \phi_{1,1}(a), \ldots \phi_{k,1}(a))$ is again a linear function. In conclusion, if $\tilde{G}$ computes G with one intermediate variable, every function in G is linear.

The concept of intermediate variables is mainly useful because of the following theorem.

Theorem 3.2.3. *Let $\tilde{G}$ compute G with n intermediate variables, n being some natural number. If $\tilde{G}$ has solution set components bound B, so has G.*

Lemma 3.2.4. *Let $f_1, \ldots, f_k, \phi$ be real-valued C^1 functions, each defined on an open subset of $\mathbb{R}^d$, and let $\tilde{f}_1, \ldots, \tilde{f}_k$ be real-valued C^1 functions on open subsets of $\mathbb{R}^{d+1}$. Suppose $\bigcap_{i=1}^k f_i^{-1}(\{0\}) \subseteq \operatorname{dom} \phi$. Also suppose that for all $a \in \operatorname{dom} \phi$ and $i = 1, \ldots, k$,*

$$\begin{aligned} a \in \operatorname{dom} f_i &\iff (a, \phi(a)) \in \operatorname{dom} \tilde{f}_i, \\ f_i(a) &= \tilde{f}_i(a, \phi(a)). \end{aligned}$$

Let $g \in C^1(\mathbb{R}^{d+1}; \mathbb{R})$ such that for $a \in \mathbb{R}^d$, $b \in \mathbb{R}$,

$$g(a, b) = 0 \iff a \in \operatorname{dom} \phi \wedge b = \phi(a) \tag{12}$$

and

$$\frac{\partial}{\partial b} g(a, \phi(a)) \neq 0, \qquad a \in \operatorname{dom} \phi.$$

Defining

$$\begin{aligned} Z &:= \bigcap_{i=1}^k f_i^{-1}(\{0\}), \\ \tilde{Z} &:= g^{-1}(\{0\}) \cap \bigcap_{i=1}^k \tilde{f}_i^{-1}(\{0\}), \end{aligned}$$

it holds that $CC(Z) = CC(\tilde{Z})$. In addition, for $a \in Z$, the Jacobian of $(f_1, \ldots, f_k)$ at a has rank k if and only if the Jacobian of $(\tilde{f}_1, \ldots \tilde{f}_k, g)$ at $(a, \phi(a))$ has rank $k + 1$.

Proof. Consider the map

$$\psi := \begin{cases} Z & \to \quad \tilde{Z}, \\ a & \mapsto \quad (a, \phi(a)). \end{cases}$$

This map is well-defined because of (12), and since $Z \subseteq \operatorname{dom} \phi$. For $(a, b) \in \tilde{Z}$, we have $g(a, b) = 0$ and further $b = \phi(a)$, which together with $\tilde{f}_i(a, \phi(a)) = 0$, $i = 1, \ldots, k$, implies $a \in \operatorname{dom} f_i$ and $f_i(a) = 0$. Consequently, $\psi^{-1}(a, b) = a$ from $\tilde{Z}$ to Z is well defined, and (12) shows that it is indeed the inverse to ψ. Hence ψ is a homeomorphism, which implies $CC(Z) = CC(\tilde{Z})$. For the second claim set $f := (f_1, \ldots f_k, g)$ and $\tilde{f} := (\tilde{f}_1, \ldots, \tilde{f}_k)$. Fix $a \in Z$. In what follows, we will consider the Jacobians of f at a and of $\tilde{f}$ and g at $((a, \phi(a))$, which are all well-defined. To simplify notation, we will just write $df, d\tilde{f}, dg$. Furthermore, we will write $d_b g$ for $\frac{\partial g}{\partial b}$ and $d_b \tilde{f}$ for $\frac{\partial \tilde{f}}{\partial b}$. We can write

$$d(\tilde{f}, g) = \begin{pmatrix} d_a f & d_a g \\ d_b \tilde{f} & d_b g \end{pmatrix}.$$

Since $d_b g \neq 0$, we can write

$$\begin{pmatrix} I_k & 0 \\ -\frac{d_b \tilde{f}}{d_b g} & 1 \end{pmatrix},$$

which is a regular $(k+1) \times (k+1)$ matrix. Therefore we have

$$\begin{aligned}
\operatorname{rank} d(\tilde{f}, g) &= \operatorname{rank}\left(\begin{pmatrix} d_a \tilde{f} & d_a g \\ d_b \tilde{f} & d_b g \end{pmatrix}\right) \\
&= \operatorname{rank}\left(\begin{pmatrix} d_a \tilde{f} & d_a g \\ d_b \tilde{f} & d_b g \end{pmatrix}\begin{pmatrix} I_k & 0 \\ -\frac{d_b \tilde{f}}{d_b g} & 1 \end{pmatrix}\right) \\
&= \operatorname{rank}\left(\begin{pmatrix} d_a \tilde{f} - \frac{d_b \tilde{f}}{d_b g} d_a g & d_a g \\ 0 & d_b g \end{pmatrix}\right) \\
&= \operatorname{rank}\left(d_a \tilde{f} - \frac{d_b \tilde{f}}{d_b g} d_a g\right) + 1.
\end{aligned}$$

Now, $0 = \frac{d}{da} g(a, \phi(a)) = d_a g + d_b g \cdot d\phi$, therefore $d\phi = -\frac{d_a g}{d_b g}$, which implies

$$\begin{aligned}
df = \frac{d}{da} \tilde{f}(a, \phi(a)) &= d_a \tilde{f} + d_b \tilde{f} \cdot d\phi \\
&= d_a \tilde{f} - d_b \tilde{f} \cdot \frac{d_a g}{d_b g}.
\end{aligned}$$

In conclusion, the rank of $d(\tilde{f}, g)$ is exactly one more than the rank of df, as required. □

With the help of this lemma, we are now able to prove Theorem 3.2.3.

Proof of Theorem 3.2.3. Let

$$S_1 := \{f_1, \ldots f_k\} \subseteq G$$

be a set with regular zero-set intersections. Let

$$S_2 := \{\tilde{f}_1, \ldots, \tilde{f}_k, g_{1,1}, \ldots g_{k,n}\}$$

be the corresponding functions according to Definition 3.2.1. Define

$$\begin{aligned}\phi_{i,j\to j}(a,b) &:= b_{i,j},\\ \phi_{i,j\to k+1}(a,b) &:= \phi_{i,k+1}(a,b_{i,1},\dots,b_{i,j},\phi_{i,j\to j+1}(a,b),\dots,\phi_{i,j\to k}(a,b)),\end{aligned}$$

for all a and b where this is well-defined, which is the set

$$\begin{aligned}\{a,b \in \bigcap_{\ell=j+1}^{k} \operatorname{dom}\phi_{i,j\to\ell} :\\ (a,b_{i,1},\dots,b_{i,j},\phi_{i,j\to j+1}(a,b),\dots,\phi_{i,j\to k}(a,b)) \in \operatorname{dom}\phi_{i,k+1}\}.\end{aligned}$$

This function represents computing intermediate variables with indices from $j+1$ to $k+1$. Now we want to change the functions in S_1 one by one. Let $q_i(j) := \lfloor \frac{j+i-1}{k} \rfloor$, or any other integer valued function with the property that $q_i(0) = 0$, $q_i(j+1) = q_i(j)+1$ for exactly one i, $q_i(j+1) = q_i(j)$ for all other i and $q_i(dk) = d$ for all i. Define also for $i = 1,\dots,k$ and $j = 0,\dots,n$ the helper functions

$$\begin{aligned}f_i^j(a,b) := \tilde{f}_i(a, b_{1,1},\dots,b_{1,q_1(j)},&\phi_{1,q_1(j)\to q_1(j)+1}(a,b),\dots,\phi_{1,q_1(j)\to n}(a,b), &(13)\\ \dots,b_{k,1},\dots,b_{k,q_k(j)},&\phi_{k,q_k(j)\to q_k(j)+1}(a,b),\dots,\phi_{k,q_k(j)\to n}(a,b)). &(14)\end{aligned}$$

This means that for the first $q_1(j)$ variables we take the input value to the function, and the subsequent ones are calculated using the ϕ functions. This definition implies $f_i^{kn} = \tilde{f}_i$, since $q_i(dk) = d$ for all i. It also holds that $f_i^0 = f_i$. To see this, define $x_{i,j} := \phi_{i,0\to j}(a,b)$ and $x := (x_{i,j})_{i=1,\dots,k,j=1,\dots,n}$. Then $x_{i,j} = \phi_{i,j}(a,x_{i,0},\dots,x_{i,j-1})$ and therefore $f_i(a) = \tilde{f}_i(a,x)$ by Definition 3.2.1 and $\tilde{f}_i(a,x) = f_i^0(a,b)$ by (13).

Now define

$$G_j := \{f_1^j, f_2^j,\dots,f_k^j, g_{1,1},\dots,g_{1,q_1(j)},\dots,g_{k,1},\dots,g_{k,q_k(j)}\}, \qquad j = 0,\dots,kd,$$

where we view all these functions as functions in the $j+d$ variables

$$a_1,\dots,a_d,b_{1,1},\dots,b_{1,q_1(j)},\dots,b_{k,1},\dots,b_{k,q_k(j)}.$$

We denote the vector of these variables as v_j. By the above considerations, $G_0 = S_1$ and $G_{kd} = S_2$. We want to show that S_1 and S_2 have the same number of solution set components. To this end we want to apply Lemma 3.2.4 to G_j and G_{j+1}. We denote by ℓ the index where $q_\ell(j+1) = q_\ell(j)+1$. Then the functions in G_{j+1} are in the variables v_{j+1}, which is the same as $(v_j, b_{\ell,q_\ell(j+1)})$. $g_{\ell,q_\ell(j+1)}$ and $\phi_{\ell,q_\ell(j+1)}$ will take on the roles of g and ϕ, and we will denote them in this way from now on forward. If $f_i^j(v_{j+1}) = 0$, then in particular $\phi_{i,q_1(j)}$ is defined. From the definition it follows from induction that

$$\phi_{\ell,q_\ell(j)\to m}(v_j,\phi(v_j)) = \phi_{\ell,q_\ell(j)+1\to m}(v_j,\phi(v_j)), \qquad m \geq q_\ell(j)+1,$$

and therefore $f_i^j(v_j) = f_i^{j+1}(v_j,\phi(v_j))$, in the sense that the left side is defined iff the right side is. Also clearly $g_{i,j}(v_j) = g_{i,j}(v_j,\phi(v_j))$ for all $g_{i,j}$ that are in G_j. Definition 3.2.1 grants the rest of the requirements. Lemma 3.2.4 shows that the zero-set of G_{j+1} has the same number of components as the zero-set of G_j. If we also assume that the differential of

$$f_1^j, f_2^j,\dots,f_k^j, g_{1,1},\dots,g_{1,q_1(j)},\dots,g_{k,1},\dots,g_{k,q_k(j)}$$

has full rank, we obtain that

$$f_1^j, f_2^j, \dots, f_k^j, g_{1,1}, \dots, g_{1,q_1(j+1)}, \dots, g_{k,1}, \dots, g_{k,q_k(j+1)}$$

has full rank. Doing this step kn times, we have shown that the zero-sets of S_2 and S_1 have the same number of components. The zero-set intersections in S_1 are regular, which implies the same for S_2. The zero-set of S_2 has at most B components by assumption, which now also holds for S_1, which finishes the proof.

□

Now we present the main result of the section: the components bound for the circuits. A circuit computes its result in multiple steps, where the result of each step can be interpreted as an intermediate variable. For that we need the following function class.

Lemma 3.2.5. *Let $f_1, \dots, f_q$ be fixed affine functions $\mathbb{R}^d \to \mathbb{R}$. Let G be the class of polynomials in $a_1, \dots, a_d, e^{f_1(a)}, \dots, e^{f_q(a)}$ with degree at most ℓ. Then G has solution set components bound*

$$B = 2^{q(q-1)/2}(\ell+1)^{2d+q}(d+1)^{d+2q}.$$

The proof for Lemma 3.2.5 can be found in [Kho91, Section 3.14, Corollary 3].

Lemma 3.2.6. *Consider a circuit satisfying the following conditions: The circuit contains q gates, the output gate computes a rational function of degree no more than $\ell > 1$, each non-output gate computes the exponential function of a rational function of degree no more than ℓ, and the denominator of each rational function is never zero. Let G be the class of functions defined on $\mathbb{R}^d$, that this circuits computes for each possible combination of eligible rational functions. Then G has solution set components bound $2^{(qd)^2/2}(9qd\ell)^{5qd}$.*

Proof. Let F be the class of polynomials in the variables $a_i, b_{i,j}, c_{i,j}$ and $e^{c_{i,j}}$ ($i = 1, \dots, d$ and $j = 1, \dots, q$) with degree at most $\ell + 1$. We will now show that F computes G with $2q - 1$ intermediate variables. To this end, fix functions $f_1, \dots, f_k$, $k \leq d$, in G. Per definition, for each function f_i and each gate j, there exist polynomials $n_{i,j}$ and $d_{i,j}$ ($i = 1, \dots, k$, $j = 1, .., q$), such that

$$v_{i,j}(a) = \exp\left(\frac{n_{i,j}(a, v_{i,1}(a), \dots, v_{i,j-1}(a))}{d_{i,j}(a, v_{i,1}(a), \dots, v_{i,j-1}(a))}\right) \tag{15}$$

and

$$f_i(a) = \frac{n_{i,q}(a, v_{i,1}(a), \dots, v_{i,q-1}(a))}{d_{i,q}(a, v_{i,1}(a), \dots, v_{i,q-1}(a))}.$$

$v_{i,j}$ denotes the output of gate j in the circuit of function f_i.

Now we define $\phi_{i,j}(a,b,c) := \left(\frac{n_{i,j}(a,b_{i,1},\dots,b_{i,j-1})}{d_{i,j}(a,b_{i,1},\dots,b_{i,j-1})}\right)$ on the sets $O_{i,j} := \{(a,b,c) \in \mathbb{R}^{d(2q+1)} : d_{i,j}(a,b,c) \neq 0\}$, and $\phi'(a,b,c) := e^{c_{i,j}}$. These will serve the function of the maps $\phi_{i,j}$ of Definition 3.2.1. If $c_{i,j} = \phi_{i,j}(a,b,c)$ and $b_{i,j} = \phi'_{i,j}(a,b,c)$ for all i, j, then clearly $c_{i,j}$ is the expression inside the exponential in (15) and $b_{i,j} = v_{i,j}(a)$, and so in that case $f_i(a) = \phi_{i,q}(a,b,c)$. Consider the functions

$$\begin{aligned}
\tilde{f}_i(a,b,c) &:= c_{i,q}, \\
g_{i,j}(a,b,c) &:= c_{i,j}d_{i,j}(a, b_{i,1}, \dots, b_{i,j-1}) - n_{i,j}(a, b_{i,1}, \dots, b_{i,j-1}), \\
h_{i,j}(a,b,c) &:= e^{c_{i,j}} - b_{i,j}, \qquad j = 1, \dots, q-1,
\end{aligned}$$

which will serve as our implicitly defining functions. Clearly these are in F, since $d_{i,j}$ has degree at most ℓ. To check item 3 in Definition 3.2.1, suppose that $(a, b, c) \in O_{i,j}$ and $c_{i,j} = \phi_{i,j}(a, b, c)$, $b_{i,j} = \phi'_{i,j}(a, b, c)$ for $j = 1, \dots, \ell - 1$. We then have that $d_{i,\ell}(a, b, c) \neq 0$ (per assumption, the denominator is never 0). This implies $(a, b, c) \in O_{i,\ell}$ and $g_{i,\ell}(a, b, c) = 0 \Leftrightarrow \phi_{i,\ell}(a, b, c) = c_{i,\ell}$ and $h_{i,\ell}(a, b, c) = 0 \Leftrightarrow \phi'_{i,\ell}(a, b, c) = b_{i,\ell}$. Finally,

$$\frac{\partial g_{i,\ell}}{\partial c_{i,\ell}}(a, \phi_{i,1}(a, b), \dots, \phi_{i,\ell}(a, b)) = d_{i,j}(a, \phi_{i,1}(a, b), \dots, \phi_{i,\ell}(a, b)) \neq 0,$$
$$\frac{\partial h_{i,\ell}}{\partial b_{i,\ell}}(a, \phi_{i,1}(a, b), \dots, \phi_{i,\ell}(a, b)) = 1 \neq 0.$$

Using Lemma 3.2.5, we get the solution set components bound

$$B = 2^{((q-1)d)((q-1)d-1)/2}(\ell + 2)^{4dq+(q-1)d}(2qd + 1)^{2qd+2(q-1)d} \tag{16}$$
$$\leq 2^{(qd)^2/2}(9qd\ell)^{5qd}. \tag{17}$$

for F. (We have substituted $(q - 1)d$ for q, $qd(q - 1)d + d$ for d and $\ell + 1$ for ℓ.) Theorem 3.2.3 implies that this is also a bound for G. □

4 Analyzing Different Network Architectures

The results from the previous section now allow us to find bounds on the VC-dimensions of a whole range of different network architectures that use sigmoid activation functions. Our approach follows [AB99, Theorem 8.13], which establishes a VC-dimension bound for standard sigmoid networks and is described here in Theorem 4.1.2. On this basis, we then prove bounds for Liquid Time Constant networks (Theorem 4.2.4) and CT-RNNs (Theorem 4.3.2). First we cite a quite general Theorem. The proof can be found in [AB99, Theorem 8.14].

Theorem 4.0.1. *Let $h\colon \mathbb{R}^d \times \mathbb{R}^n \to \{0, 1\}$, determining the class*

$$H := \{x \mapsto h(a, x)\}.$$

Suppose that h can be computed by an algorithm that takes $(a, x) \in \mathbb{R}^d \times \mathbb{R}^n$ as input and outputs $h(a, x)$ after no more than t of the following operations:

- *addition, subtraction, multiplication and devision on real numbers,*
- *jumps conditioned on comparisons using $>, \geq, <, \leq, =, \neq$ on real numbers,*
- *the exponential function $x \mapsto e^x$*
- *output 0 or 1.*

Then

$$\mathrm{VCdim}(H) \leq t^2 d(d + 19 log_2(9d)).$$

In principle, for each of the network types we consider we could use this theorem to find a VC-Dimension bound. We will, however, use a more direct approach in the rest of this section, where we will need the following lemma.

Lemma 4.0.2. *Suppose $2^m \leq Cm^b$, then $m \leq 2\log_2(C) + b\log_2(\frac{b}{\ln 2})$.*

Proof. First we will prove the inequality $\ln(a) \leq ab + \ln(1/b) - 1$ for $a > 0, b > 0$. By looking at the derivative, one can deduce $e^x \geq 1 + x$ for $x \in \mathbb{R}$. This implies $e^{ab-1} \geq ab$ and subsequently $\ln ab \leq ab - 1$.

Using this inequality, we obtain

$$\log_2(m) \leq \frac{m}{2b} + \log_2\left(\frac{b}{\ln 2}\right)$$

and further

$$\begin{aligned} 2^m \leq Cm^b &\Longrightarrow m \leq \log_2(C) + b\left(\frac{m}{2b} + \log_2\left(\frac{b}{\ln 2}\right)\right) \\ &\Longleftrightarrow m \leq 2\left(\log_2(C) + b\log_2\left(\frac{b}{\ln 2}\right)\right). \end{aligned}$$

□

4.1 Classical Feed-Forward Network

First we take a look at a classical feed-forward network. This architecture is one of the oldest neural-network architectures and the most straight-forward. The value of a neuron is determined by the value of a certain collection of other neurons. This is shown in Figure 4.1, where the black lines represent which neurons take input from which. Oftentimes the neurons are organized in layers, such as in the figure, but it does not nessecarily have to be the case. The only restriction of the connections is that there are no loops, so no neuron is indirectly dependent on its own value. We now give a formal definition of feed-forward networks.

Definition 4.1.1. *For our purposes, we define a standard sigmoid network in the following way. Let d denote the number of input neurons and k the number of non-input neurons. These neurons are represented by the functions $x_1, \ldots, x_k$, which will be defined below. Let $C \subseteq \{1, \ldots, k\}^2$, where $i < j$ for every $(i,j) \in C$, denote the "connection matrix". A pair (i,j) being in C means that neuron j takes input from neuron i. The restriction $i < j$ means that neurons only take inputs from previous neurons. Similarly, $D \subseteq \{1, \ldots, d\} \times \{1, \ldots, k\}$ denotes which neuron takes input from which input unit. For every $\tilde{w} = (w_{i,j})_{(i,j)\in C} \in \mathbb{R}^C$, $w' = (w'_{i,j})_{(i,j)\in D} \in \mathbb{R}^D$ (weights) and $b = (b_i)_{i=1}^k \in \mathbb{R}^k$ (biases), we write $w = (\tilde{w}, w', b)$ and define*

$$x_j(w,a) := \sigma\left(b_j + \sum_{i:(i,j)\in C} w_{i,j}x_i + \sum_{i:(i,j)\in D} w'_{i,j}a_i\right), \qquad j = 1, \ldots, k-1,$$

and

$$x_k(w,a) = b_j + \sum_{i:(i,k)\in C} w_{i,k}x_i + \sum_{i:(i,k)\in D} w'_{i,k}a_i,$$

where $\sigma(t) := \frac{1}{1+e^t}$. The number $x_j(w,a)$ is the value of the jth neuron of a network with weights and biases w, that is given the vector a $(\in \mathbb{R}^d)$ as input. x_k represents the output neuron, so the associated binary classifier is given by $(a \mapsto \operatorname{bsgn}(x_k(w,a)))$.

Theorem 4.1.2. *Let H be the set of functions computed by a standard feed-forward sigmoid-network as in Definition 4.1.1 with W parameters (weights and biases) and $k-1$ computation units. Then*

$$\Pi_H(m) \leq 2^{(Wk)^2/2}(18Wk^2)^{5Wk}\left(\frac{em}{W}\right)^W$$

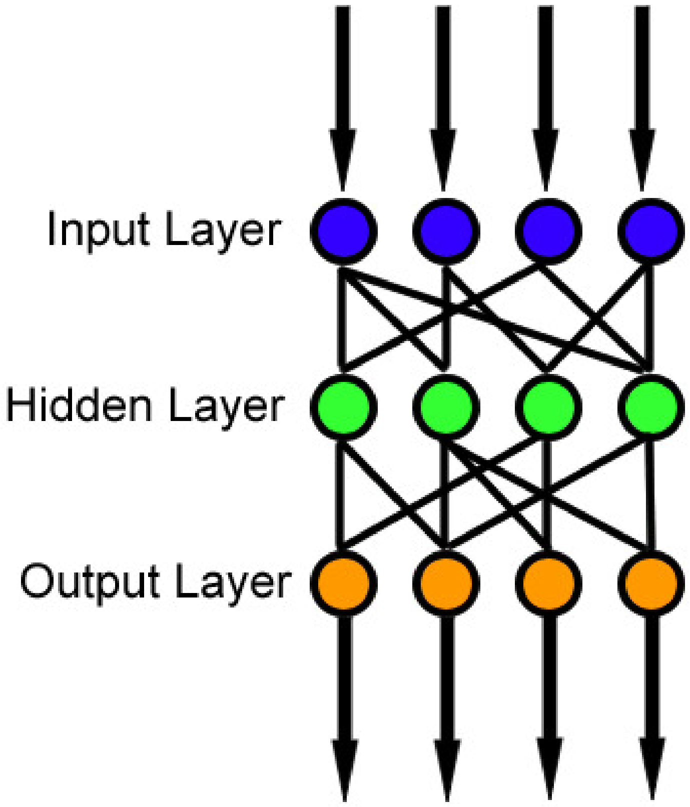

Figure 1: A feed-forward neural network.

(provided $m \geq W$) and subsequently

$$\mathrm{VCdim}(H) \leq (Wk)^2 + 11Wk \log_2(18Wk^2).$$

Proof. Let H be the class of classifiers of Definition 4.1.1. For every such classifier h, there is $w \in \mathbb{R}^W$, such that $h(a) = \mathrm{bsgn}(x_k(w,a))$, which means that H is a 1-combination of $\{x_k\}$ (where we view $x_k : \mathbb{R}^W \times \mathbb{R}^d \to \mathbb{R}$). Following Definition 3.1.9 and Remark 3.1.11, we consider the class of functions $\{(w \mapsto x_j(w,a) + c, a \in \mathbb{R}^d, c \in \mathbb{R}\}$. We will respresent the computation of the network as a circuit of the kind in Lemma 3.2.6. To do this, we split the computation of the sigmoid function into the computation of $(t \mapsto e^t)$ and $(t \mapsto \frac{1}{1+t})$, which will be done by different circuits. So in the setting of the previous definition, define $y_j := \frac{1}{x_j} - 1$ (notice that per definition $x_j \neq 0$). Then

$$\begin{aligned} y_j(w,a) &= \exp\left(b_j + \sum_{i:(i,j)\in C} w_{i,j} x_i(w,a) + \sum_{i:(i,j)\in D} w'_{i,j} a_i\right) \\ &= \exp\left(b_j + \sum_{i:(i,j)\in C} w_{i,j} \frac{1}{1+y_i(w,a)} + \sum_{i:(i,j)\in D} w'_{i,j} a_i\right) \end{aligned}$$

for $j = 1,\ldots,k$ and

$$x_k(w,a) + c = \frac{1}{1+y_k(w,a)} + c.$$

So for every input $a \in \mathbb{R}^d$, the values $y_1,\ldots,y_k, x_k + c$ are computed by a circuit of the kind in Lemma 3.2.6 with $k+1$ gates. (Note: According to [AB99], only $k-1$ gates are needed. We believe this to be an error. However, it does not substantially affect the rest of this theses.) The first sum is a sum of at most $k-1$ numbers, since each summand corresponds to a neuron. Therefore, the expression inside the exponential can be factorized into a rational function in $(w, y_1(a),\ldots,y_k(a))$ of degree at most k. Since functions of the form $x_k(.,a)+c$ are a subset of all possible circuits of the size, Lemma 3.2.6 shows that $\{x_k + c : c \in \mathbb{R}\}$ has solution set components bound

$$2^{(kW)^2/2}(9k^2W)^{5kW}$$

with respect to the first d variables. Now we use Theorem 3.1.10 to obtain

$$\Pi_H(m) \leq 2^{(kW)^2/2}(9k^2W)^{5kW} \sum_{i=0}^{W} \binom{m}{i} \leq 2^{(kW)^2/2}(9k^2W)^{5kW}(em/W)^W, \tag{18}$$

where the last inequality is because of (2). If $\Pi_H(m) \geq 2^m$, then by Lemma 4.0.2 we have

$$m \leq (kW)^2 + 10kW\log_2(9k^2W\ell) + 2W(\log_2(e) - \log_2(W)) + W\log_2(\frac{W}{\ln 2}).$$

Therefore $\mathrm{VCdim}(H) \leq (kW)^2 + 11kW\log_2(9k^2W)$.

□

4.2 Liquid Time Constant Networks

Liquid Time Constant networks are a new neural network algorithm. They described for example in [HLA⁺20]. It is a form of NeuralODE, which means that each neuron has a value that is dependent on time (where time goes from 0 to some positive value T) and these values are described by an ordinary differential equation (namely (19) and (20)). Its design is inspired by biological neural networks.

Definition 4.2.1. *An LTC network is a structure made up of n neurons, each of which stores a value that is dependent on time. We call this value $x_j(t)$, where $j \in \{1,\ldots,n\}$ is the index of the neuron and $t \in [0,T]$ is the time. Now in an LTC network, these functions satisfy the following set of equations (where (19) is differential).*

$$C x'_i(t) = w_{li}(e_{li} - x_i(t)) + \sum_{j=1}^{n} y_{ji}(t), \tag{19}$$

$$y_{ji}(t) = w_{ji}\sigma(v_{ji}(x_j(t) + \mu_{ji}))(e_{ji} - x_i(t)). \tag{20}$$

In addition, if $a = (a_1, \dots, a_n)$ is the vector of inputs, then

$$x_i(0) = a_i. \tag{21}$$

The classifier to a set of parameters w is then defined as $h(a) := \operatorname{bsgn}(x_n(T))$, and $x_1, \dots, x_n$ is the unique solution to the initial value problem (19) – (21).

Remark 4.2.2. This initial value problem does in fact have a unique global solution by the Picard-Lindelöf theorem.

Definition 4.2.3. *Both in our theoretical considerations and in practice we use a discretized version of the ODE. We use a form of hybrid implicit/explicit one-step method, introduced in [HLA⁺20]. For the integration times $0 = t_0 < t_1 < \ldots < t_N = T$ define*

$$x_i(t_{k+1}) = \frac{x_i(t_k)C_i/\delta_k + w_{li}e_{li} + \sum_{j=1}^n w_{ji}\sigma(x_j(t_k), \mu_j)e_{ji}}{C_i/\delta_k + w_{li} + \sum_{j=1}^n w_{ji}\sigma(x_j(t_k), \mu_j)}, \tag{22}$$

where $\delta_k = t_{k+1} - t_k$.

Theorem 4.2.4. *Let H be the class of classifiers produced by a discretized LTC network with W weights, n neurons and N integration steps, where $n, N \geq 4.L$ Then $VCdim(H) \leq (nNW)^2 + 11nNW \log_2(9(nN)^2W)$.*

Proof. As in Theorem 4.1.2, we will try to rewrite this equation into a circuit described in Lemma 3.2.6. For this, define again $y_i(t_k) := \exp(-x_i(t_k))$. Then we have

$$y_i(t_0) = \exp(-a_i), \tag{23}$$

$$y_i(t_{k+1}) = \exp\left(-\frac{x_i(t_k)C_i/\delta + w_{li}e_{li} + \sum_{j=1}^n \frac{w_{ji}\cdot e_j i}{1+y_j(t_k)}}{C_i/\delta + w_{li} + \sum_{j=1}^n \frac{w_{ji}}{1+y_j(t_k)}}\right), \qquad k = 0, \dots, N-2, \tag{24}$$

$$x_n(t_N) = \frac{x_n(t_{N-1})C_i/\delta + w_{li}e_{li} + \sum_{j=1}^n \frac{w_{ji}\cdot e_{ji}}{1+y_j(t_{N-1})}}{C_i/\delta + w_{li} + \sum_{j=1}^n \frac{w_{ji}}{1+y_n(t_{N-1})}}. \tag{25}$$

Here we drop the dependency of y_i and x_i on the parameters w and inputs a. The remaining difficulty lies in describing the term $x_i(t_k)$. We can see by an inductive argument, using the fact that

$$\begin{aligned} x_i(t_{k+1}) &= \frac{x_i(t_k)C_i/\delta + w_{li}e_{li} + \sum_{j=1}^n \frac{w_{ji}\cdot e_{ji}}{1+y_i(t_k)}}{C_i/\delta + w_{li} + \sum_{j=1}^n \frac{c_5}{1+y_i(t_k)}} \\ &= \frac{\prod_{\ell=1}^n (1 + y_\ell(t_k))(x_i(t_k)C_i/\delta + w_{li}e_{li}) + \sum_{j=1}^n \prod_{\substack{1\leq \ell \leq n \\ \ell \neq j}} (1 + y_\ell(t_k))w_{ji} \cdot e_{ji}}{\prod_{\ell=1}^n (1 + y_\ell(t_k))C_i/\delta + w_{li} + \sum_{j=1}^n \prod_{\substack{1\leq \ell \leq n \\ \ell \neq j}} (1 + y_\ell(t_k))w_{ji}}, \end{aligned} \tag{26}$$

that $x_i(t_k) = r_{i,k}(w, y(t_0), \dots, y(t_{k-1}))$, where r is a rational function. The degrees of these rational functions satisfy the inequality

$$\deg r_{i,k+1} \leq \deg r_{i,k} + 1 + 2 + n + 3n - 3 + 2n - 2 \leq \deg r_{i,k} + 7n - 1.$$

Since deg $r_{i,0} = 0$, we get $\deg r_{i,k} \leq 7nk - 1$. Therefore the degree inside the exponential is at most $7nN + 3n + 2 \leq 8nN$ (since $n, N \geq 4$). Equations (23) – (25) define a circuit with $Nn + 1$ gates, so Lemma 3.2.6 gives

$$\Pi_H(m) \leq 2^{(nNW)^2/2}(72nNWnN)^{5nNW}(em/W)^W. \tag{27}$$

This inequality holds due to the slightly lower bound in (16) compared to the formulation of the lemma. Again using Lemma 4.0.2, $2^m = \Pi_H(m)$ implies

$$m \leq (nNW)^2 + 10nNW \log_2(72nNWnN) + 2W \log_2(\frac{e}{W}) + W \log_2(W),$$

from which

$$\mathrm{VCdim}(H) \leq (nNW)^2 + 11nNW \log_2(72(nN)^2W)$$

follows. □

4.3 Continuous Time Recurrent Neural Networks

Continuous time recurrent neural networks, also known as CT-RNNs, are also a form of NeuralODE. They have been around for a little longer than LTCs, and are very similar. The defining differential equation is a little bit different.

Definition 4.3.1. *A CT-RNN network is a structure made of n neurons with time-dependent values, which will be denoted as $x_j(t)$, similar to the LTC networks of Definition 4.2.1. A CT-RNN network is defined by the equation*

$$x_i'(t) = -\tau_i x_i + \sum_j a_{i,j}\sigma(v_{i,j}(x_j + \mu_{i,j})), \qquad i = 1, \ldots, n.$$

For $i, j = 1, \ldots, n$, the variables τ_i,$a_{i,j}$, $v_{i,j}$, $\mu_{i,j}$ are learnable parameters of the network.

We discretize the equation in a hybrid manner as

$$x_i(t_{k+1}) = x_i(t_k) - \tau_i x_i(t_{k+1}) + \sum_j a_{i,j}\sigma(v_{i,j}(x_j(t_k) + \mu_{i,j})),$$

which we can rewrite as

$$x_i(t_{k+1}) = \left(x_i(t_k) + \sum_j a_{i,j}\sigma(v_{i,j}(x_j(t_k) + \mu_{i,j})) \right) \frac{1}{1+\tau_i}.$$

As with LTCs, we consider the binary classifier $h(a) = \mathrm{bsgn}(x_n(T))$.

Theorem 4.3.2. *The set of classifiers produced by the discretized version of a CT-RNN with n neurons, W parameters and N integration steps has a VC dimension of at most $(nNW)^2 + 11nNW \log_2(9nNW(n+1)(N+1))$.*

Proof. Define again $y_i(t_k) := \exp(-v_{i,j}(x_j(t_k) + \mu_{i,j}))$ such that

$$y_i(t_{k+1}) = \exp\left(-v_{i,j} \left(x_i(t_k) + \sum_j a_{i,j} \frac{1}{1+y_i(t_k)} + \mu_{i,j} \right) \frac{1}{1+\tau_i} \right).$$

Again, we need to express $x_i(t_k)$ in terms of a polynomial in $y_i(t_k)$ and the parameters. For this we use the identity

$$x_i(t_{k+1}) = \left(x_i(t_k) + \sum_j a_{i,j} \frac{1}{1+y_j(t_k)} \right) \frac{1}{1+\tau_i}.$$

We see that the degree of the rational function increases by at most $n+1$ each step, so the degree is at most $(n+1)N$. From this we can conclude that the rational functions inside the exponential have a degree of at most $(n+1)N+n+1$. Analogous to the proof of Theorem 4.2.4, Lemma 3.2.6 and Theorem 3.1.10 give

$$\Pi_H(m) \leq 2^{(nNW)^2/2}(9nNW(n+1)(N+1))^{5nNW}(em/W)^W. \tag{28}$$

Using Lemma 4.0.2 we get for $m \leq \mathrm{VCdim}(H)$

$$\begin{aligned} m \leq (nNW)^2 &+ 10nNW\log_2(9nNW(n+1)(N+1)) \\ &+ 2W\log_2(\frac{e}{W}) + W\log_2(W), \end{aligned}$$

in other words,

$$\mathrm{VCdim}(H) \leq (nNW)^2 + 11nNW\log_2(9nNW(n+1)(N+1)).$$

□

As one can see, the estimate for the two network types is virtually identical. An LTC network with equal size will be a little more powerful due to the fact that it has more parameters.

4.4 Summary of Bounds

The following table summarizes the results of this section.

Network type	Proven bound	Appr. bound
Feed-forward	$(Wk)^2 + 11Wk\log_2(18Wk^2)$	$(nW)^2/2$
CT-RNN	$(nNW)^2 + 11nNW\log_2(9nNW(n+1)(N+1))$	$(nNW)^2/2$
LTC	$(nNW)^2 + 11nNW\log_2(9(nN)^2W)$	$(nNW)^2/2$

The approximate bounds are derived from equations (18), (27) and (28), since for large n, N, W all terms except $2^{(nW)^2}$ or $2^{(nNW)^2}$ are relatively small.

4.5 Comparing the Network Types

In this section, we will look at some examples with concrete numbers that compare the VC-dimension bounds of the different network types. The bounds were not calculated with neither proven bound nor approximate bound from the table above, but instead by finding numerically the highest value m, such that equations (18), (27) or (28) are satisfied when substituting 2^m for $\Pi_H(m)$. The number of parameters for the NeuralODE models is caculated as

$$\begin{aligned} \mathrm{param}_{\mathrm{LTC}}(n) &= 4n(n-1) + 3n, \\ \mathrm{param}_{\mathrm{CTRNN}}(n) &= 3n(n-1) + n, \end{aligned}$$

where n is the number of neurons. For the feed-forward network, we used the formula

$$\mathrm{param}_{\mathrm{FFW}}(w, d, i) = wd + wi + w^2(d-1),$$

where w is the number of neurons per layer ("width"), d is the number of layers except for the input layer ("depth") and i is the number of input neurons. The networks considered are built in such a way that $w = 10d$ (for simplicity it was not ensured that w and d are integers). The

Integration steps	LTC ($\times 10^9$)	CT-RNN ($\times 10^9$)
10	13.023	6.922
20	52.059	27.711
30	117.108	62.366
40	208.168	110.888
50	325.240	173.277

Table 1: VC Dimension of LTC and CT-RNN with 16 neurons.

Integration steps	LTC ($\times 10^9$)	CT-RNN ($\times 10^9$)
10	845.781	463.155
20	3 382.828	1 852.819
30	7 611.131	4 169.000
40	13 530.686	7 411.700
50	21 141.492	11 580.919

Table 2: VC Dimension of LTC and CT-RNN with 32 neurons.

number of inputs is always equal to the number of neurons of the neuralODE that it is being compared to. We made this choice since this will ensure that the compared networks both have the same number of inputs.

First we will have a look at LTCs and CT-RNNs. Since the bound formula is virtually the same for both types, the difference in expressiveness at the same size comes from the fact that the LTC has more parameters per neuron. In Tables 1, 2 and 3 one finds a numerical comparison of these values. As one can see, the VC dimension of the LTC is about double that of the CT-RNN. The VC-dimension increases rapidly with the number of integration steps, and even more rapidly with the number of neurons. This is because the bound approximately increases by a power of 6 with the number of neurons (note that more neurons means more parameters) and only quadratically with the number of integration steps. The same tendency can also be seen in Figure 2. The type of model used only makes a small difference for the bound compared to the number of integration steps and neurons.

Next, we will have a look at classical feed-forward networks and compare them to LTC models. The comparison to CT-RNNs is omitted since we have already seen that there is no big difference to LTCs. In Figure 4.5 one can see how large a classical neural net needs to be to have the same VC dimension bound as an LTC network, dependent on number of neurons and integration steps. There seems to be a almost linear dependence between the two. Looking a bit closer, Figure 5 shows the ratio of the size of a classical net and an LTC. As one can see, the ratio does actually

Integration steps	LTC ($\times 10^{15}$)	CT-RNN ($\times 10^{15}$)
10	3.505	1.959
20	14.019	7.834
30	31.542	17.627
40	56.075	31.337
50	87.618	48.964

Table 3: VC Dimension of LTC and CT-RNN with 128 neurons.

increase with size (and of course the number of integration steps of the LTC). For 20 integration steps, a small LTC has the same VC dimension bound as a classical neural network only 5 times the size, while for a 1000 neuron LTC one would already need a feed-forward network with 10 000 neurons. The difference is larger in the case of 500 integration steps, where a 50 times larger classical net is needed.

Lastly, we look at the effect of the structure of the feed-forward network. In Figure 6 we can see how big a classical neural network needs to be to match the VC dimension bound of a given LTC. The different lines represent different ratios of width and depth. For the ratio 5 for example, this means that there are 5 times more neurons in each layer than there are layers. We find a sizeable effect that makes the VC dimension bound of wider networks higher as compared to deeper networks. That is because the number of parameters increases faster with width than with depth. In Figure 7 we find a variation to Figure 5. Here, we fix the depth of the network to 5 layers. We see that in this case, the LTC really is just a constant factor larger than the classical network. The classical network is made larger only by increasing the number of neurons per layer, which makes the number of parameters grow faster than increasing the number of layers.

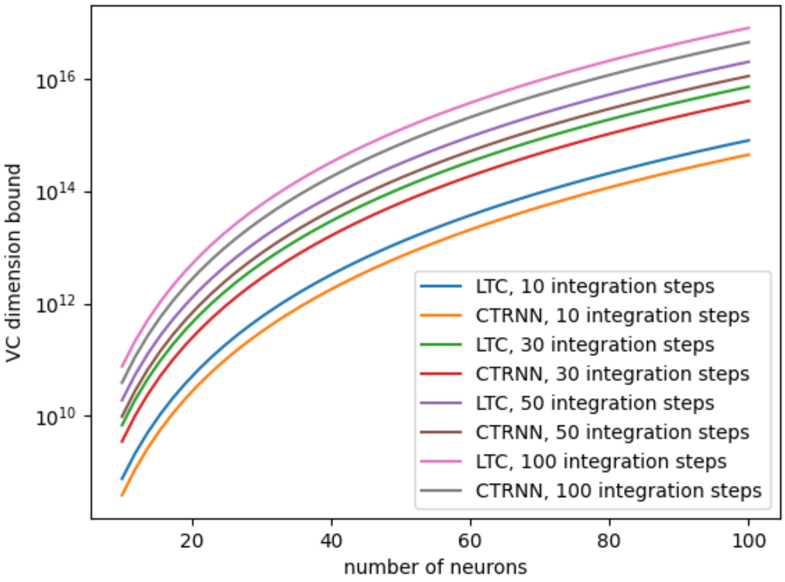

Figure 2: Comparison of LTC and CT-RNN

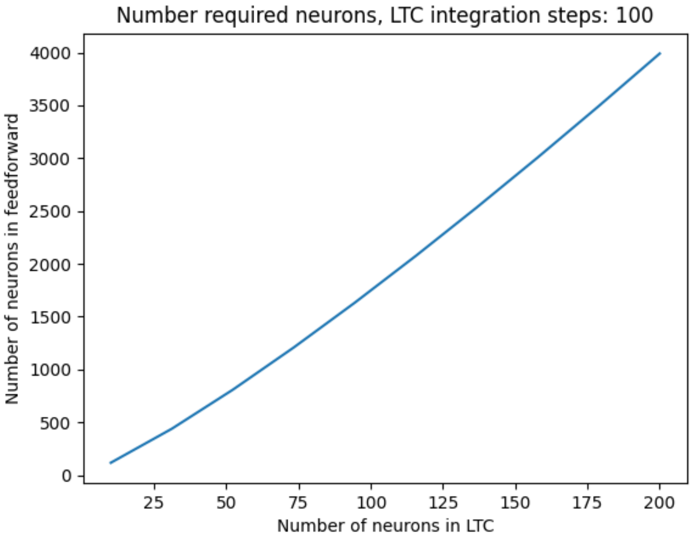

Figure 3: Comparing feed-forward and LTC networks.

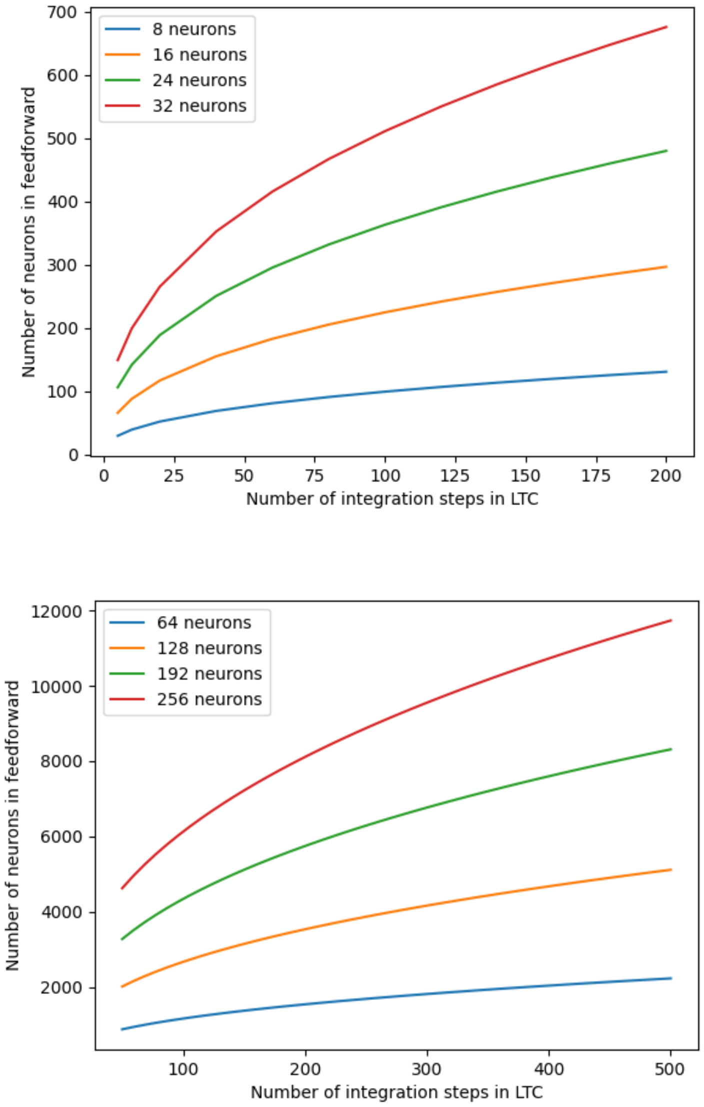

Figure 4: Size of equivalent classical NN dependent on LTC integration steps

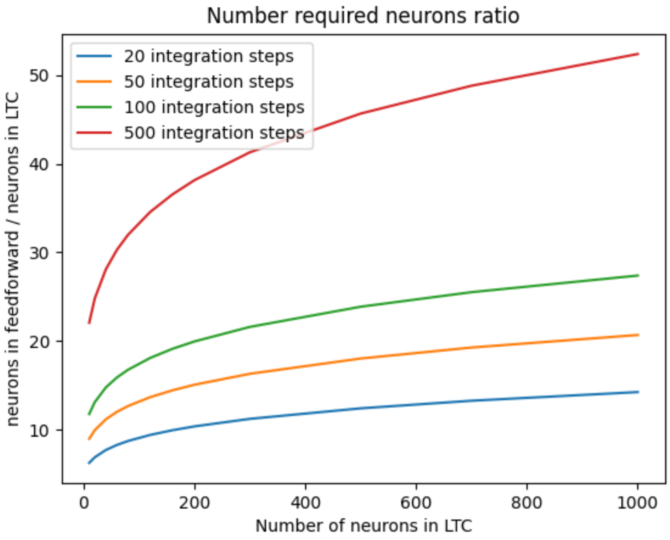

Figure 5: Ratio of sizes of classical/LTC networks

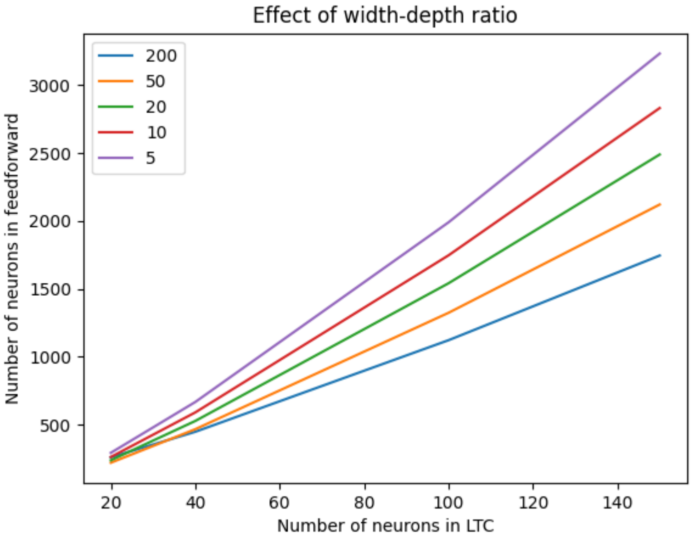

Figure 6: Effect of width-depth ratio

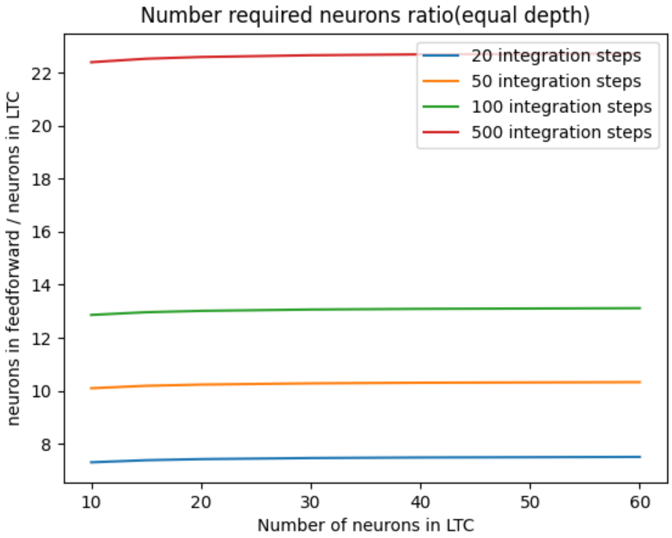

Figure 7: Size ratio of equivalent classical network with fixed depth and LTC

5 Discussion

Our results showcase the important variables in determining the VC-dimension of neural networks. These are the number of neurons, the number of parameters, and in case of recurrent model types, the number of integration steps. One potential upside to the recurrent types is that you can have a much bigger VC-dimension with fewer neurons. This is because the number of integration steps increases the VC-dimension, and also because these architectures typically have a lot more parameters per neuron. Arguably, this is helpful for the interpretability of these networks for the following reasons. Oftentimes, it is possible to understand the behaviour of a single neuron, even if it is very complex. This in turn makes it is easier to understand networks that have fewer neurons. In other words, a neuralODE with the same number of neurons as a regular neural net is more powerful, while still having the same interpretability.

In our approach of looking at the VC-dimension of recurrent models we treat these models as a binary classifier, of which the output is based on the value of a specific neuron at a specific time. This is oftentimes not the way they are used in practice. For example, if used for automatic parking, the value of multiple neurons and at every timestep determines the path that the car takes. This discrepancy needs to be taken into account when wanting to estimate the expressiveness of these networks in the different use cases. It is to be assumed though, that a higher power as a classifier also suggests a higher power in other situations.

Two further questions come to mind. Firstly, how accurate are the bounds to the VC-dimension? This is of course hard to answer. But it depends on how tight the bounds are in Theorem 3.1.10, in Theorem 3.2.3, in Lemma 3.2.5 and also in the final estimates of Theorems 4.1.2, 4.2.4 and 4.3.2. Note that in these final estimates we used the VC-dimension bound for the function class of all circuits and did not take into account that only a small subset of these circuits actually respresents the computations of a neural network. Also, for feed-forward networks, the discrepancy between the best known upper bound ($\mathcal{O}((Wk)^2)$, see Theorem 4.1.2) and lower bound ($\Omega(W^2)$, see [AB99, Theorem 8.9]) is still very large.

Secondly, is the VC-dimension even a suitable tool for measuring the expressiveness of a neural network? If a network has VC dimension n, then there are n points in the input space such that the network can express every possible classification of these points. So it is a measure of how complicated the classifications that the network expresses can be. It can be argued that a high VC-dimension also suggests that the network will be able to be more expressive in a situation where it is not used as a classifier.

It shall be noted at this point, that a high VC-dimension is not always preferable. Higher VC-dimension is generally associated with more difficult learning. In [AB99, Theorem 4.2] one can find such a result. It gives an estimation of the learning success when drawing a sample from any distribution. It is dependent on the VC-dimension of the function class containing the possible classifiers, and the bound only works if the size of the sample is at least half the VC-dimension. One remarkable thing about neural networks is that much smaller sample sizes are oftentimes sufficient for good learning success.

References

[AB99] Martin Anthony and Peter L. Bartlett. *Neural Network Learning: Theoretical Foundations.* Cambridge University press, 1999.

[Ano] Anonymous. Wikipedia: History of artificial neural networks. https://en.wikipedia.org/wiki/History_of_artificial_neural_networks.

[BH89] Eric B. Baum and David Haussler. What size net gives valid generalization? *Neural Computation*, 1:151–160, March 1989.

[HLA$^+$20] Ramin M. Hasani, Mathias Lechner, Alexander Amini, Daniela Rus, and Radu Grosu. Liquid time-constant networks. *Computing Research Repository*, abs/2006.04439, 2020.

[Kal14] Michael Kaltenbäck. *Fundament Analysis.* Berliner Buchreihe zur Mathematik. Heldermann Verlag, 2014.

[Kal18] Michael Kaltenbäck. Analysis 3 für Technische Mathematik, 2018. Lecture Notes, TU Wien.

[Kho91] A.G. Khovanskii. *Fewnomials*, volume 88 of *Translation of Mathematical Monographs.* American Mathematical Society, 1991.

[KM97] Marek Karpinski and Angus Macintyre. Polynomial bounds for VC Dimension of sigmoidal and general pfaffian neural networks. *Journal of Computer and System Sciences*, 54:169–176, 1997.

[Ste64] Shlomo Sternberg. *Lectures on Differential Geometry.* Prentice-Hall, 1964.

[VC71] V. N. Vapnik and A. Ya. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability & Its Applications*, 16(2):264–280, 1971.

[War68] Hugh E. Warren. Lower bounds for approximation by nonlinear manifolds. *Transactions of the American Mathematical Society*, 133(1):167–178, August 1968.