Performance of (deep) learning models

Franck Iutzeler

These notes heavily build on the course as given by François Bachoc until 2025
https://francoisbachoc.github.io/Lecture_notes_theory_deep_learning.pdf

Version: July 9, 2026

Contents

CHAPTER 1 INTRODUCTION

THE purpose of this first part is to properly introduce the notations and the notions of machine learning theory upon which the course will be based.

With deep learning, we shall understand neural networks with many hidden layers. Deep learning methods are currently very popular for some tasks, for instance the following ones.

- **regression:** predicting $y \in \mathbb{R}$.
- **classification:** predicting $y \in \{0, 1\}$ or $y \in \{0, \dots, K\}$.
- **generative modeling:** generating vectors $x \in \mathbb{R}^d$ following an unknown target distribution.

Typical applications are:

- **For regression:** any type of input $x \in \mathbb{R}^d$ and of corresponding output $y \in \mathbb{R}$ to predict. For instance, y can be the delay of a flight and x can gather characteristics of this flight, such as the day, position of the airport and duration.
- **For classification:** $x \in \mathbb{R}^d$ can be an image (vector of color levels for each pixels) and y can give the type of image, for instance cat/dog, or value of a digit.
- **For generative modeling:** generating images (e.g. faces) or musical pieces.

Goals of the lecture notes. The goal is to study some theoretical aspects of deep learning, and in some cases of machine learning more broadly. There are many recent contributions and only a few of them will be covered.

1.1 Regression

We consider a law $\mathcal{L}$ on $[0,1]^d \times \mathbb{R}$. We aim at finding a function $f : [0,1]^d \to \mathbb{R}$ such that, for $(X, Y) \sim \mathcal{L}$,

$$\mathbb{E}\left((f(X) - Y)^2\right)$$

is small.

The optimal function f is then the conditional expectation, as shown in the following proposition.

Proposition 1.1. *Let $f^\star : [0,1]^d \to \mathbb{R}$ be defined by*

$$f^\star(x) = \mathbb{E}\left(Y | X = x\right),$$

for $x \in [0,1]^d$. Then, for any $f : [0,1]^d \to \mathbb{R}$,

$$\mathbb{E}\left((f(X)-Y)^2\right) = \mathbb{E}\left(\left(f^\star(X)-Y\right)^2\right) + \mathbb{E}\left(\left(f^\star(X)-f(X)\right)^2\right).$$

From the previous proposition, $f^\star$ minimizes the mean square error among all possible functions, and the closer a function f is to $f^\star$, the more it leads to a small mean square error.

Proof of Proposition 1.1 Let us use the law of total expectation.

$$\mathbb{E}\left((f(X)-Y)^2\right) = \mathbb{E}\left(\mathbb{E}\left(\left(f(X)-Y\right)^2 \middle| X\right)\right).$$

Conditionally to X, we can use the equation

$$\mathbb{E}\left((Z-a(X))^2 | X\right) = \mathrm{Var}(Z|X) + \left(\mathbb{E}(Z|X) - a(X)\right)^2$$

for a random variable Z and a function $a(X)$ (bias-variance decomposition). This gives

$$\begin{aligned}
\mathbb{E}\left((f(X)-Y)^2\right) =& \mathbb{E}\left(\left(\mathbb{E}(Y|X)-f(X)\right)^2 + \mathrm{Var}(Y|X)\right)\\
=& \mathbb{E}\left(\left(f^\star(X)-f(X)\right)^2\right) + \mathbb{E}\left(\mathrm{Var}(Y|X)\right)\\
=& \mathbb{E}\left(\left(f^\star(X)-f(X)\right)^2\right) + \mathbb{E}\left(\mathbb{E}\left(\left(Y-\mathbb{E}(Y|X)\right)^2 \middle| X\right)\right)\\
\text{(law of total expectation :)} =& \mathbb{E}\left(\left(f^\star(X)-f(X)\right)^2\right) + \mathbb{E}\left(\left(Y-\mathbb{E}(Y|X)\right)^2\right)\\
=& \mathbb{E}\left(\left(f^\star(X)-f(X)\right)^2\right) + \mathbb{E}\left(\left(Y-f^\star(X)\right)^2\right).
\end{aligned}$$

□

We now consider a data set of the form $(X_1,Y_1),\dots,(X_n,Y_n)$, independent and of law $\mathcal{L}$. We consider a function learned by *empirical risk minimization*. We let $\mathcal{F}$ be a set of functions from $[0,1]^d$ to $\mathbb{R}$. We consider

$$\hat{f}_n \in \operatorname*{argmin}_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^n \left(f(X_i)-Y_i\right)^2.$$

The next proposition enables to bound the mean square error of $\hat{f}_n$.

Proposition 1.2. *Let $(X,Y) \sim \mathcal{L}$, independently from $(X_1,Y_1),\dots,(X_n,Y_n)$. Then we have*

$$\begin{aligned}
&\mathbb{E}\left(\left(\hat{f}_n(X)-Y\right)^2\right) - \mathbb{E}\left(\left(f^\star(X)-Y\right)^2\right)\\
&\le 2\mathbb{E}\left(\sup_{f\in\mathcal{F}}\left|\mathbb{E}\left(\left(f(X)-Y\right)^2\right) - \frac{1}{n}\sum_{i=1}^n \left(f(X_i)-Y_i\right)^2\right|\right) + \inf_{f\in\mathcal{F}} \mathbb{E}\left(\left(f(X)-f^\star(X)\right)^2\right).
\end{aligned}$$

Remarks

- In the term
$$\mathbb{E}\left(\left(\hat{f}_n(X)-Y\right)^2\right)$$
the expectation is taken with respect to both $(X_1,Y_1),\dots,(X_n,Y_n)$ and (X,Y).
- We bound
$$\mathbb{E}\left(\left(\hat{f}_n(X)-Y\right)^2\right) - \mathbb{E}\left(\left(f^\star(X)-Y\right)^2\right)$$
which is always non-negative and is called the *excess of risk*.

- The first component of the bound is
$$2\mathbb{E}\left(\sup_{f\in\mathcal{F}}\left|\mathbb{E}\left((f(X)-Y)^2\right)-\frac{1}{n}\sum_{i=1}^{n}(f(X_i)-Y_i)^2\right|\right)$$
which is called the *generalization error*. The larger the set $\mathcal{F}$ is, the larger this error is, because the supremum is taken over a larger set.
- The second component of the bound is
$$\inf_{f\in\mathcal{F}}\mathbb{E}\left(\left(f(X)-f^\star(X)\right)^2\right)$$
which is called the *approximation error*. The smaller $\mathcal{F}$ is the larger this error is, because the infimum is taken over less functions.
- Hence, we see that $\mathcal{F}$ should be not too small and not too large, which can be interpreted as a bias-variance trade off.

Proof of Proposition 1.2
We let, for $f\in\mathcal{F}$,
$$R(f)=\mathbb{E}\left((Y-f(X))^2\right)$$
and
$$R_n(f)=\frac{1}{n}\sum_{i=1}^{n}(f(X_i)-Y_i)^2.$$
We let, for $\epsilon>0$, f_ϵ be such that
$$R(f_\epsilon)\leq\inf_{f\in\mathcal{F}}R(f)+\epsilon.$$
Then we have
$$\begin{aligned}
&\mathbb{E}\left(\left(\hat{f}_n(X)-Y\right)^2\right)\\
\text{(law of total expectation:)}\quad=&\mathbb{E}\left(\mathbb{E}\left(\left(\hat{f}_n(X)-Y\right)^2\middle|X_1,Y_1,\ldots,X_n,Y_n\right)\right)\\
=&\mathbb{E}\left(R(\hat{f}_n)\right),
\end{aligned}$$
since (X,Y) is independent from $X_1,Y_1,\ldots,X_n,Y_n$ and in $R(\hat{f}_n)$, the function $\hat{f}_n$ is fixed, as the expectation is taken only with respect to X and Y. Then we have
$$\begin{aligned}
\mathbb{E}\left(R(\hat{f}_n)\right)-R(f^\star)=&\mathbb{E}\left(R(\hat{f}_n)-R_n(\hat{f}_n)\right)+\mathbb{E}\left(R_n(\hat{f}_n)-R_n(f_\epsilon)\right)+\mathbb{E}\left(R_n(f_\epsilon)-R(f_\epsilon)\right)\\
&+\left(R(f_\epsilon)-\inf_{f\in\mathcal{F}}R(f)\right)+\left(\inf_{f\in\mathcal{F}}R(f)-R(f^\star)\right)\\
\leq&\mathbb{E}\left(\sup_{f\in\mathcal{F}}|R(f)-R_n(f)|\right)+0+\mathbb{E}\left(\sup_{f\in\mathcal{F}}|R(f)-R_n(f)|\right)\\
&+\epsilon+\inf_{f\in\mathcal{F}}\left(R(f)-R(f^\star)\right)\\
\text{(Proposition 1.1:)}\quad=&2\mathbb{E}\left(\sup_{f\in\mathcal{F}}|R(f)-R_n(f)|\right)+\epsilon+\inf_{f\in\mathcal{F}}\mathbb{E}\left(\left(f(X)-f^\star(X)\right)^2\right).
\end{aligned}$$
Since this inequality holds for any $\epsilon>0$, we also obtain the inequality with $\epsilon=0$ which concludes the proof. □

1.2 Classification

The general principle is quite similar to regression. We consider a law $\mathcal{L}$ on $[0,1]^d \times \{0,1\}$. We are looking for a function $f : [0,1]^d \to \{0,1\}$ (a *classifier*) such that with $(X,Y) \sim \mathcal{L}$,

$$\mathbb{P}\left(f(X) \neq Y\right)$$

is small. The next proposition provides the optimal function f for this.

Proposition 1.3. *Let $p^\star : [0,1]^d \to [0,1]$ defined by*

$$p^\star(x) = \mathbb{P}\left(Y = 1 | X = x\right)$$

for $x \in [0,1]^d$. We let

$$T^\star(x) = \mathbf{1}_{p^\star(x) \geq \frac{1}{2}}$$

for $x \in [0,1]^d$. Then, for any $f : [0,1]^d \to \{0,1\}$,

$$\mathbb{P}\left(f(X) \neq Y\right) = \mathbb{P}\left(T^\star(X) \neq Y\right) + \mathbb{E}\left(\mathbf{1}_{T^\star(X) \neq f(X)} \left|1 - 2p^\star(X)\right|\right).$$

Hence, we see that a prediction error (that is, predicting $f(X)$ with $f(X) \neq T^\star(X)$) is more harmful when

$$\left|1 - 2p^\star(X)\right|$$

is large. This is well interpreted, because when

$$\left|1 - 2p^\star(X)\right| = 0,$$

we have $p^\star(X) = 1/2$, thus $\mathbb{P}(Y = 1|X) = 1/2$. In this case, $\mathbb{P}(f(X) \neq Y|X) = 1/2$, regardless of the value of $f(X)$.

Proof of Proposition 1.3 Using the law of total expectation, we have

$$\begin{aligned}\mathbb{P}\left(f(X) \neq Y\right) - \mathbb{P}\left(T^\star(X) \neq Y\right) &= \mathbb{E}\left(\mathbb{E}\left(\mathbf{1}_{f(X) \neq Y} - \mathbf{1}_{T^\star(X) \neq Y} \middle| X\right)\right) \\ &:= \mathbb{E}\left(\mathbb{E}\left(e(X,Y) | X\right)\right).\end{aligned}$$

Conditionally to X we have the following.

- If $T^\star(X) = 1$, then
  - if $f(X) = 1$, then $e(X,Y) = 0$,
  - if $f(X) = 0$, then
    - $e(X,Y) = 1$ with probability $\mathbb{P}(Y = 1|X) = p^\star(X)$,
    - $e(X,Y) = -1$ with probability $\mathbb{P}(Y = 0|X) = 1 - p^\star(X)$

  and thus

  $$\mathbb{E}\left(e(X,Y) | X\right) = \mathbf{1}_{f(X) \neq T^\star(X)} \left(p^\star(X) - \left(1 - p^\star(X)\right)\right) = \mathbf{1}_{f(X) \neq T^\star(X)} \left|1 - 2p^\star(X)\right|.$$

- If $T^\star(X) = 0$, then
  - if $f(X) = 0$, then $e(X,Y) = 0$,
  - if $f(X) = 1$, then
    - $e(X,Y) = 1$ with probability $\mathbb{P}(Y = 0|X) = 1 - p^\star(X)$,
    - $e(X,Y) = -1$ with probability $\mathbb{P}(Y = 1|X) = p^\star(X)$

and thus

$$\mathbb{E}\left(e(X,Y)|X\right) = \mathbf{1}_{f(X)\neq T^\star(X)}\left(1-p^\star(X)-p^\star(X)\right) = \mathbf{1}_{f(X)\neq T^\star(X)}\left|1-2p^\star(X)\right|.$$

Hence, eventually

$$\mathbb{P}\left(f(X)\neq Y\right) - \mathbb{P}\left(T^\star(X)\neq Y\right) = \mathbb{E}\left(\mathbf{1}_{T^\star(X)\neq f(X)}\left|1-2p^\star(X)\right|\right).$$

□

We now consider a data set of the form $(X_1,Y_1),\dots,(X_n,Y_n)$ independent and of law $\mathcal{L}$. We consider a function that is learned by empirical risk minimization. We consider a set $\mathcal{F}$ of functions from $[0,1]^d$ to $\{0,1\}$ and

$$\hat{f}_n \in \operatorname*{argmin}_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}_{Y_i\neq f(X_i)}.$$

The next proposition enables to bound the probability of error of $\hat{f}_n$.

Proposition 1.4. *Let $(X,Y)\sim\mathcal{L}$, independently from $(X_1,Y_1),\dots,(X_n,Y_n)$. Then we have*

$$\begin{aligned}&\mathbb{P}\left(\hat{f}_n(X)\neq Y\right) - \mathbb{P}\left(T^\star(X)\neq Y\right)\\ \leq& 2\mathbb{E}\left(\sup_{f\in\mathcal{F}}\left|\mathbb{P}\left(f(X)\neq Y\right) - \frac{1}{n}\sum_{i=1}^{n}\mathbf{1}_{f(X_i)\neq Y_i}\right|\right)\\ &+ \inf_{f\in\mathcal{F}}\mathbb{E}\left(\mathbf{1}_{T^\star(X)\neq f(X)}\left|1-2p^\star(X)\right|\right).\end{aligned}$$

The proof and the interpretation are the same as for regression.

1.3 Neural networks

Neural networks define a set of functions from $[0,1]^d$ to $\mathbb{R}$.

Feed-forward neural networks with one hidden layer This is the simplest example. These networks are represented as in Figure 1.1.

In Figure 1.1, the interpretation is the following.

- The arrows mean
  - that there is a multiplication by a scalar
  - or that a function from $\mathbb{R}$ to $\mathbb{R}$ is applied and (possibly) a scalar is added.
- The function $\sigma:\mathbb{R}\to\mathbb{R}$ is called the *activation function.*
- A circle (a neuron) sums all the values that are pointed to it by the arrows.
- The column with $w_1,\dots,w_N$ is called the hidden layer.

The function corresponding to Figure 1.1 is

$$x\in[0,1]^d \mapsto \sum_{i=1}^{N} v_i\sigma\left(\langle w_i,x\rangle + b_i\right),$$

with $\langle\cdot,\cdot\rangle$ the standard inner product on $\mathbb{R}^d$.

The neural network function is parametrized by

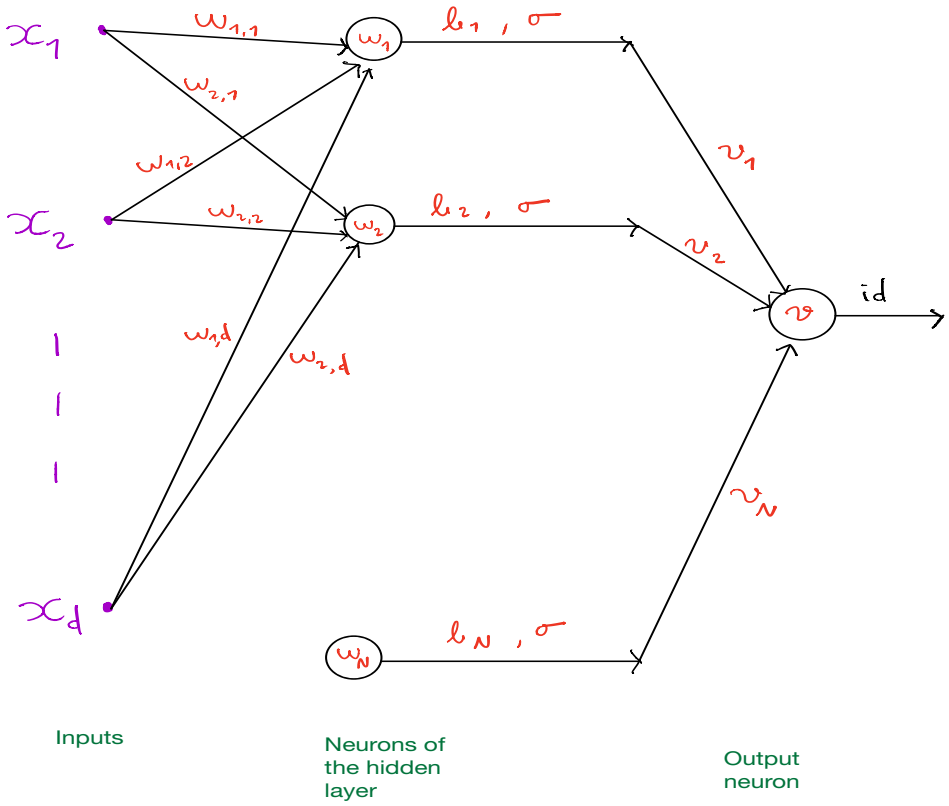

Figure 1.1: Representation of a feed-forward neural network with one hidden layer.

- $\sigma : \mathbb{R} \to \mathbb{R}$, the activation function,
- $v_1, \ldots, v_N \in \mathbb{R}$, the *output weights*,
- $w_1, \ldots, w_N \in \mathbb{R}^d$, the *weights (of the neurons of the hidden layer)*,
- $b_1, \ldots, b_N \in \mathbb{R}$, the *biases*.

Examples of activation functions are, for $t \in \mathbb{R}$,

- *linear* $\sigma(t) = t$,
- *threshold* $\sigma(t) = \mathbf{1}_{t \geq 0}$,
- *sigmoid* $\sigma(t) = e^t/(1+e^t)$,
- *ReLU* $\sigma(t) = \max(0, t)$.

For instance, when $d = 1$, the network of Figure 1.2 encodes the absolute value function with σ the ReLU function.

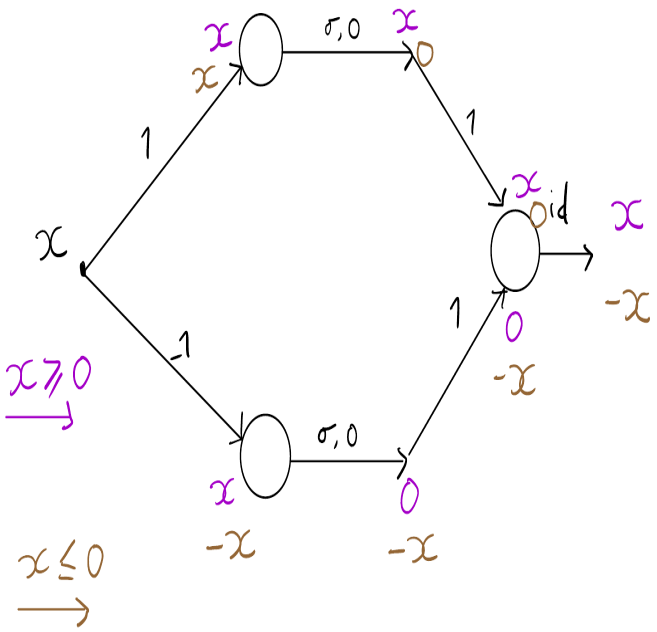

Figure 1.2: Representation of the absolute value function as a neural network.

Feed-forward neural networks with several hidden layers. This is the same type of representation but with several layers of activation functions. These networks are represented as in Figure 1.3.

The neural network function corresponding to Figure 1.3 is defined by

$$x \in [0,1]^d \mapsto f_v \circ g_c \circ g_{c-1} \circ \cdots \circ g_1(x), \tag{1.1}$$

where

$$f_v : \mathbb{R}^{N_c} \to \mathbb{R}$$
$$u \to \sum_{i=1}^{N_c} u_i v_i$$

and for $i = 1, \ldots, c$, with $N_0 = d$,

$$g_i : \mathbb{R}^{N_{i-1}} \to \mathbb{R}^{N_i}$$

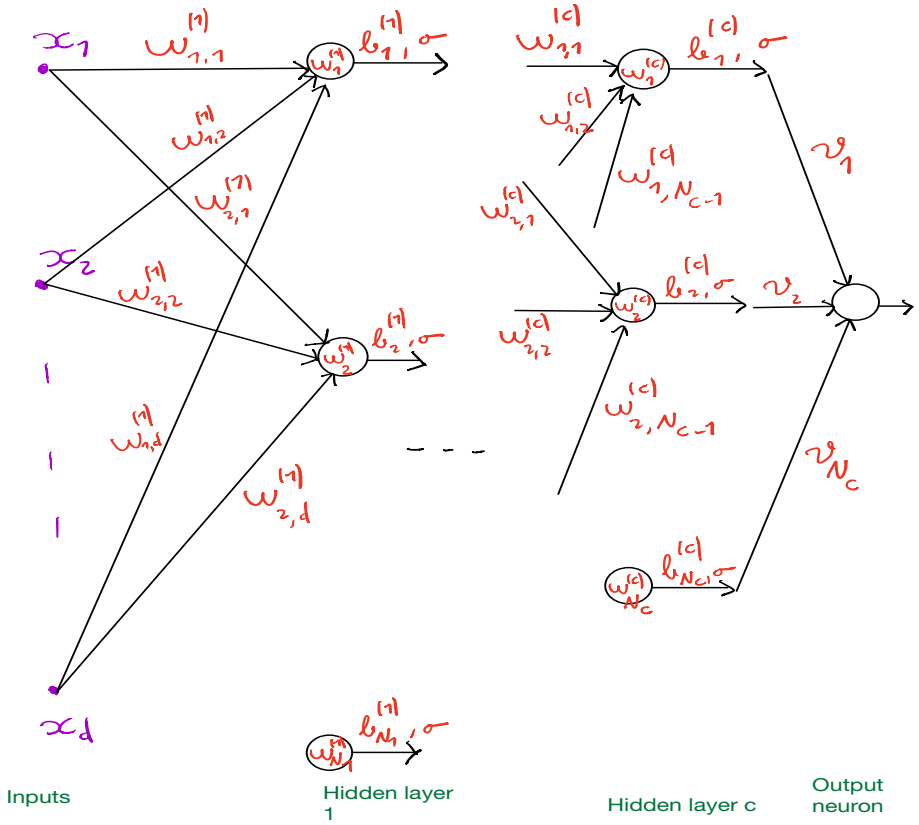

Figure 1.3: Representation of a feed-forward neural network with several hidden layer.

is defined by, for $u \in \mathbb{R}^{N_{i-1}}$ and $j = 1, \dots, N_i$,

$$(g_i(u))_j = \sigma\left(\langle w_j^{(i)}, u\rangle + b_j^{(i)}\right).$$

The neural network function is parametrized by

- $\sigma : \mathbb{R} \to \mathbb{R}$, the activation function,
- $v \in \mathbb{R}^{N_c}$, the *output weights*,
- $b_1^{(c)}, \dots, b_{N_c}^{(c)} \in \mathbb{R}$, the *biases of the hidden layer* c,

- $w_1^{(c)}, \dots, w_{N_c}^{(c)} \in \mathbb{R}^{N_{c-1}}$, the *weights of the hidden layer* c,
- $\vdots$
- $b_1^{(2)}, \dots, b_{N_2}^{(2)} \in \mathbb{R}$, the *biases of the hidden layer* 2,
- $w_1^{(2)}, \dots, w_{N_2}^{(2)} \in \mathbb{R}^{N_1}$, the *weights of the hidden layer* 2,
- $b_1^{(1)}, \dots, b_{N_1}^{(1)} \in \mathbb{R}$, the *biases of the hidden layer* 1,
- $w_1^{(1)}, \dots, w_{N_1}^{(1)} \in \mathbb{R}^d$, the *weights of the hidden layer* 1.

Classes of functions

To come back to regression, the class of functions $\mathcal{F}$ corresponding to neural networks is given by

- c, number of hidden layers,
- σ, activation function,
- $N_1, \dots, N_c$, numbers of neurons in the hidden layers.

These parameters are called *architecture parameters*. Then, for a given architecture, $\mathcal{F}$ is a parametric set of functions

$$\mathcal{F} = \Big\{ \text{neural networks parametrized by } v, b_1^{(c)}, \dots, b_{N_c}^{(c)}, w_1^{(c)}, \dots, w_{N_c}^{(c)}, \dots, b_1^{(1)}, \dots, b_{N_1}^{(1)}, w_1^{(1)}, \dots, w_{N_1}^{(1)} \Big\}.$$

For classification, for $g \in \mathcal{F}$, we take

$$f(x) = \begin{cases} 1 & \text{if } g(x) \geq 0 \\ 0 & \text{if } g(x) < 0 \end{cases}$$

to have a parametric set of classifiers.

CHAPTER 2 Approximation with neural networks

The purpose of this first part is to properly introduce the notations and the notions of stochastic programming and how randomness can intervene in optimization methods.

2.1 Statement of the theorem

Several theorems tackle the *universality* of feed-forward neural networks with one hidden layer of the form

$$x \in [0,1]^d \mapsto \sum_{i=1}^{N} v_i \sigma(\langle w_i, x\rangle + b_i)$$

with $v_1, \dots, v_N \in \mathbb{R}$, $b_1, \dots, b_N \in \mathbb{R}$, $w_1, \dots, w_N \in \mathbb{R}^d$ and $\sigma : \mathbb{R} \to \mathbb{R}$.

We will study the first theorem of the literature, from (Cybenko, 1989).

Theorem 2.1 ((Cybenko, 1989)). *Let $\sigma : \mathbb{R} \to \mathbb{R}$ be a continuous function such that*

$$\begin{cases} \sigma(t) \underset{t \to -\infty}{\to} 0 \\ \sigma(t) \underset{t \to +\infty}{\to} 1. \end{cases}$$

Then the set $\mathcal{N}_1$ of functions of the form

$$x \in [0,1]^d \mapsto \sum_{i=1}^{N} v_i \sigma(\langle w_i, x\rangle + b_i);$$

$N \in \mathbb{N}, v_1, \dots, v_N \in \mathbb{R}, b_1, \dots, b_N \in \mathbb{R}, w_1, \dots, w_N \in \mathbb{R}^d$ is dense in the set $C([0,1]^d, \mathbb{R})$ of the real-valued continuous functions on $[0,1]^d$, endowed with the supremum norm, $||f||_\infty = \sup_{x \in [0,1]^d} |f(x)|$ for $f \in C([0,1]^d, \mathbb{R})$.

This theorem means the following.

- We have $\mathcal{N}_1 \subset C([0,1]^d, \mathbb{R})$, which means that neural network functions are continuous.

- For all $f \in C([0,1]^d, \mathbb{R})$, for all $\epsilon > 0$, there exist $N \in \mathbb{N}$, $v_1, \dots, v_N \in \mathbb{R}$, $b_1, \dots, b_N \in \mathbb{R}$ and $w_1, \dots, w_N \in \mathbb{R}^d$ such that

$$\sup_{x \in [0,1]^d} \left| f(x) - \sum_{i=1}^{N} v_i \sigma(\langle w_i, x \rangle + b_i) \right| \leq \epsilon.$$

- Equivalently, for all $f \in C([0,1]^d, \mathbb{R})$, for all $\epsilon > 0$, there exists $g \in \mathcal{N}_1$ such that $||f - g||_\infty \leq \epsilon$.
- Equivalently, for all $f \in C([0,1]^d, \mathbb{R})$, there exists a sequence $(g_n)_{n \in \mathbb{N}}$ such that $g_n \in \mathcal{N}_1$ for $n \in \mathbb{N}$ and $||f - g_n||_\infty \to 0$ as $n \to \infty$.
- This theorem is comforting for the approximation error

$$\inf_{f \in \mathcal{F}} \mathbb{E}\left(\left(f(X) - f^\star(X) \right)^2 \right)$$

  in regression with $\mathcal{F} = \mathcal{N}_1$. Indeed, this term is equal to zero if $x \mapsto f^\star(x)$ (conditional expectation in regression) is a continuous function on $[0,1]^d$.
- Furthermore, if we let, for $N \in \mathbb{N}$,

$$\mathcal{N}_{1,N} = \left\{ x \in [0,1]^d \mapsto \sum_{i=1}^{N} v_i \sigma(\langle w_i, x \rangle + b_i); v_1, \dots, v_N \in \mathbb{R}, b_1, \dots, b_N \in \mathbb{R}, w_1, \dots, w_N \in \mathbb{R}^d \right\}$$

  (set of neural networks with N neurons), then we remark that $\mathcal{N}_{1,k} \subset \mathcal{N}_{1,k+1}$ for $k \in \mathbb{N}$. The proof of this inclusion is left as an exercize, one can for instance construct a neural network with $k+1$ neurons and $v_{k+1} = 0$ to obtain the function of a neural network with k neurons. Hence, we have that $\inf_{f \in \mathcal{N}_{1,N}} ||f - f^\star||_\infty$ is decreasing with N. Hence, from Theorem 2.1, $\inf_{f \in \mathcal{N}_{1,N}} ||f - f^\star||_\infty \to 0$ as $N \to \infty$ (left as an exercize). Hence, since $\mathbb{E}(g(X)^2) \leq ||g||_\infty^2$ for $g : [0,1]^d \to \mathbb{R}$, we obtain

$$\inf_{f \in \mathcal{N}_{1,N}} \mathbb{E}\left(\left(f(X) - f^\star(X) \right)^2 \right) \underset{N \to \infty}{\to} 0.$$

  Hence if we minimize the empirical risk with neural networks with N neurons (N large), the approximation error will be small.

2.2 Sketch of the proof

The proof is by contradiction (it is non constructive). For $f \in C([0,1]^d, \mathbb{R})$ we will not exhibit a neural network that is close to f.

Step 1

We assume that there exists $f_0 \in C([0,1]^d, \mathbb{R}) \backslash \overline{\mathcal{N}}_1$. Here we write $\overline{\mathcal{N}}_1$ for the closure of $\mathcal{N}_1$, which means that

$$f \in \overline{\mathcal{N}}_1 \iff \text{there exists } (g_N)_{N \in \mathbb{N}} \text{ with } g_N \in \mathcal{N}_1 \text{ for } N \in \mathbb{N} \text{ such that } ||g_N - f_0||_\infty \underset{N \to \infty}{\to} 0.$$

Step 2

We apply the *Hahn-Banach theorem* to construct a continuous linear map

$$L : C([0,1]^d, \mathbb{R}) \to \mathbb{C}$$

such that $L(f_0) = 1$ and $L(f) = 0$ for all $f \in \mathcal{N}_1$.

- Linear means that for $g_1, g_2 \in C([0,1]^d, \mathbb{R})$ and for $\alpha_1, \alpha_2 \in \mathbb{R}$, we have

$$L(\alpha_1 g_1 + \alpha_2 g_2) = \alpha_1 L(g_1) + \alpha_2 L(g_2).$$

- Continuous means that for $g \in C([0,1]^d, \mathbb{R})$ and for a sequence $(g_n)_{n\in\mathbb{N}}$ with $g_n \in C([0,1]^d, \mathbb{R})$ for $n \in \mathbb{N}$ and such that $||g_n - g||_\infty \to 0$ as $n \to \infty$, we have

$$L(g_n) \to L(g)$$

as $n \to \infty$.

Step 3

We then use the *Riesz representation theorem*. There exists a complex-valued Borel measure μ on $[0,1]^d$ such that

$$L(f) = \int_{[0,1]^d} f d\mu$$

for $f \in C([0,1]^d, \mathbb{R})$, where the above integral is a Lebesgue integral. That μ is a complex-valued Borel measure on $[0,1]^d$ means that, with $\mathcal{B}$ the Borel sigma algebra (the measurable subsets of $[0,1]^d$), we have

$$\mu : \mathcal{B} \to \mathbb{C}.$$

Furthermore, for $E \in \mathcal{B}$ such that $E = \cup_{i=1}^\infty E_i$ with $E_1, E_2, \ldots \in \mathcal{B}$, with $E_i \cap E_j = \emptyset$ for $i \neq j$, we have

$$\mu(E) = \sum_{i=1}^\infty \mu(E_i),$$

where $\mu(E_i) \in \mathbb{C}$ and $\sum_{i=1}^\infty |\mu(E_i)| < \infty$.

Step 4

We show that $\int_{[0,1]^d} f d\mu = 0$ for all $f \in \mathcal{N}_1$ implies that $\mu = 0$, which is a contradiction to $L(f_0) = \int_{[0,1]^d} f_0 d\mu = 1$ and concludes the proof.

Remark 2.2. The steps 1, 2 and 3 could be carried out with $\mathcal{N}_1$ replaced by other function spaces $\mathcal{F}$. These steps are actually classical in approximation theory. The step 4 is on the contrary specific to neural networks with one hidden layer. ◂

2.3 Complete proof

Let $f \in \mathcal{N}_1$. Then there exist $N \in \mathbb{N}, v_1, \ldots, v_N \in \mathbb{R}, b_1, \ldots, b_N \in \mathbb{R}, w_1, \ldots, w_N \in \mathbb{R}^d$ such that

$$f : x \in [0,1]^d \mapsto \sum_{i=1}^N v_i \sigma(\langle w_i, x\rangle + b_i).$$

Since σ is continuous, f is continuous as a sum and composition of continuous functions. Hence $\mathcal{N}_1 \subset C([0,1]^d, \mathbb{R})$. Let us now assume that $\overline{\mathcal{N}_1} \neq C([0,1]^d, \mathbb{R})$. Thus, let $f_0 \in C([0,1]^d, \mathbb{R}) \backslash \overline{\mathcal{N}_1}$. We then apply a version of the Hahn-Banach theorem.

Theorem 2.3. *There exists a continuous linear map*

$$L : C([0,1]^d, \mathbb{R}) \to \mathbb{C}$$

such that $L(f_0) = 1$ *and* $L(f) = 0$ *for all* $f \in \mathcal{N}_1$.

The above theorem holds because $f_0 \notin \overline{\mathcal{N}}_1$, see for instance (Rudin, 1998)[Chapters 3 and 6]. We then apply a version of the Riesz representation theorem.

Theorem 2.4. *There exists a complex-valued Borel measure μ on $[0,1]^d$ such that*

$$L(f) = \int_{[0,1]^d} f d\mu$$

for $f \in C([0,1]^d, \mathbb{R})$. We have seen that $\mu : \mathcal{B} \to \mathbb{C}$ where, for $B \in \mathcal{B}$, we have $B \subset [0,1]^d$. Furthermore, we can defined the total variation measure $|\mu|$ defined by

$$|\mu|(E) = \sup \sum_{i=1}^{\infty} |\mu(E_i)|,$$

$E \in \mathcal{B}$ where the supremum is over the set of all the $(E_i)_{i\in\mathbb{N}}$, with $E_i \in \mathcal{B}$ for $i \in \mathbb{N}$ and $E_i \cap E_j = \emptyset$ for $i \neq j$ and $E = \cup_{i=1}^{\infty} E_i$. Then $|\mu| : \mathcal{B} \to [0,\infty)$ and $|\mu|$ has finite mass, $|\mu|([0,1]^d) < \infty$.
Finally, there exists $h : [0,1]^d \to \mathbb{C}$, measurable, such that $|h(x)| = 1$ for $x \in [0,1]^d$ and

$$d\mu = h d|\mu|$$

which means that for $B \in \mathcal{B}$,

$$\mu(B) = \int_B h d|\mu| = \int_B h(x) d|\mu|(x)$$

and we have a more classical Lebesgue integral with a function h that corresponds to a density.

The above theorem is also given in (Rudin, 1998). The theorem implies that $L(f) = \int_{[0,1]^d} f(x)h(x)d|\mu|(x)$ for $f \in C([0,1]^d, \mathbb{R})$.

We now want to show that $\mu = 0$, which means that $\mu(B) = 0$ for $B \in \mathcal{B}$. Since $L(f) = 0$ for all $f \in C([0,1]^d, \mathbb{R})$, we have, for all $N \in \mathbb{N}$, $v_1, \dots, v_N \in \mathbb{R}$, $b_1, \dots, b_N \in \mathbb{R}$, $w_1, \dots, w_N \in \mathbb{R}^d$, with

$$f = \sum_{i=1}^{N} v_i f_i,$$

where for $i = 1, \dots, N$, $f_i : [0,1]^d \to \mathbb{R}$ is defined by, for $x \in [0,1]^d$,

$$f_i(x) = \sigma(\langle w_i, x\rangle + b_i),$$

we have $L(f) = 0$. Hence, since L is linear

$$\sum_{i=1}^{N} v_i L(f_i) = 0.$$

Specifically, we can choose $v_1 = 1$ and $v_2 = \cdots = v_N = 0$ to obtain, for all $w \in \mathbb{R}^d$, for all $b \in \mathbb{R}$,

$$L(f) = 0,$$

with $f : [0,1]^d \to \mathbb{R}$ defined by, for $x \in [0,1]^d$,

$$f(x) = \sigma(\langle w, x\rangle + b).$$

This gives, for all $w \in \mathbb{R}^d$, for all $b \in \mathbb{R}$,

$$L(f) = \int_{[0,1]^d} f(x) d\mu(x) = 0$$

and thus

$$\int_{[0,1]^d} \sigma(\langle w, x\rangle + b) h(x) d|\mu|(x) = 0.$$

Let $w \in \mathbb{R}^d$ and $b, \phi \in \mathbb{R}$, $\lambda > 0$. Let $x \in [0,1]^d$. We let

$$\sigma_{\lambda,\phi}(x) = \sigma\left(\lambda\left(\langle w, x\rangle + b\right) + \phi\right).$$

Then if $\langle w, x\rangle + b > 0$, since $\sigma(t) \longrightarrow 1$ as $t \longrightarrow +\infty$, we have

$$\sigma\left(\lambda\left(\langle w, x\rangle + b\right) + \phi\right) \underset{\lambda \to +\infty}{\longrightarrow} 1.$$

If $\langle w, x\rangle + b < 0$, since $\sigma(t) \longrightarrow 0$ as $t \longrightarrow -\infty$, we have

$$\sigma\left(\lambda\left(\langle w, x\rangle + b\right) + \phi\right) \underset{\lambda \to +\infty}{\longrightarrow} 0.$$

If $\langle w, x\rangle + b = 0$, then we have

$$\sigma\left(\lambda\left(\langle w, x\rangle + b\right) + \phi\right) = \sigma(\phi).$$

Hence, we have shown that

$$\sigma_{\lambda,\phi}(x) \underset{\lambda \to +\infty}{\longrightarrow} \gamma(x) := \begin{cases} 1 & \text{if } \langle w, x\rangle + b > 0 \\ 0 & \text{if } \langle w, x\rangle + b < 0 \\ \sigma(\phi) & \text{if } \langle w, x\rangle + b = 0 \end{cases}.$$

Furthermore, for $x \in [0,1]^d$,

$$\sigma_{\lambda,\phi}(x) = \sigma\left(\lambda\left(\langle w, x\rangle + b\right) + \phi\right) = \sigma\left(\langle \lambda w, x\rangle + \lambda b + \phi\right)$$

and thus $\sigma_{\lambda,\phi} \in \mathcal{N}_1$ (it is a neural network function). Hence

$$\int_{[0,1]^d} \sigma_{\lambda,\phi}(x) h(x) d|\mu|(x) = 0.$$

Furthermore,

$$\sup_{\lambda > 0} \sup_{x \in [0,1]^d} |\sigma_{\lambda,\phi}(x)| \leq \sup_{t \in \mathbb{R}} \sigma(t) = ||\sigma||_\infty < \infty,$$

as σ is continuous and has finite limits at $\pm\infty$. We recall that $|h(x)| = 1$ for all $x \in [0,1]^d$ and thus

$$\int_{[0,1]^d} \sup_{\lambda > 0} |\sigma_{\lambda,\phi}(x)||h(x)| d|\mu|(x) \leq \left(\sup_{t \in \mathbb{R}} |\sigma(t)|\right) \int_{[0,1]^d} d|\mu|(x) = \left(\sup_{t \in \mathbb{R}} |\sigma(t)|\right) |\mu|([0,1]^d) < \infty.$$

Hence we can apply the dominated convergence theorem,

$$\begin{aligned} 0 &= \int_{[0,1]^d} \sigma_{\lambda,\phi}(x) h(x) d|\mu|(x) \underset{\lambda \to +\infty}{\longrightarrow} \int_{[0,1]^d} \gamma(x) h(x) d|\mu|(x) \\ &= \int_{[0,1]^d} \left(\mathbf{1}_{\langle w,x\rangle + b > 0} + \sigma(\phi) \mathbf{1}_{\langle w,x\rangle + b = 0}\right) h(x) d|\mu|(x). \end{aligned}$$

We let

$$\Pi_{w,b} = \left\{x \in [0,1]^d : \langle w, x\rangle + b = 0\right\}$$

and

$$H_{w,b} = \left\{x \in [0,1]^d : \langle w, x\rangle + b > 0\right\}$$

for $w \in \mathbb{R}^d$ and $b \in \mathbb{R}$. We then obtain

$$\int_{H_{w,b}} h(x)d|\mu|(x) + \sigma(\phi)\int_{\Pi_{w,b}} h(x)d|\mu|(x) = 0$$

and thus

$$\mu(H_{w,b}) + \sigma(\phi)\mu(\Pi_{w,b}) = 0.$$

Since σ is not constant, we can take $\phi_1, \phi_2 \in \mathbb{R}$ with $\sigma(\phi_1) \neq \sigma(\phi_2)$ and thus

$$\begin{pmatrix} 1 & \sigma(\phi_1) \\ 1 & \sigma(\phi_2) \end{pmatrix}\begin{pmatrix} \mu(H_{w,b}) \\ \mu(\Pi_{w,b}) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

and the determinant of the above matrix is $\sigma(\phi_1) - \sigma(\phi_2) \neq 0$. Hence, for all $w \in \mathbb{R}^d$ and $b \in \mathbb{R}$,

$$\mu(\underbrace{\Pi_{w,b}}_{\text{hyperplane}}) = \mu(\underbrace{H_{w,b}}_{\text{half space}}) = 0.$$

Let $w \in \mathbb{R}^d$. We write $||w||_1 = \sum_{i=1}^d |w_i|$. For a bounded $g : [-||w||_1, ||w||_1] \to \mathbb{C}$ (not necessarily continuous), we let

$$\psi(g) = \int_{[0,1]^d} g(\langle w, x\rangle)d\mu(x).$$

We remark that

$$|\langle w, x\rangle| = \left|\sum_{i=1}^d w_i x_i\right| \leq \sum_{i=1}^d |w_i| = ||w||_1.$$

We observe that ψ is linear, for any bounded $g_1, g_2 : [-||w||_1, ||w||_1] \to \mathbb{C}$, for any $\alpha_1, \alpha_2 \in \mathbb{R}$, we have

$$\begin{aligned}
\psi(\alpha_1 g_1 + \alpha_2 g_2) &= \int_{[0,1]^d} \left(\alpha_1 g_1(\langle w, x\rangle) + \alpha_2 g_2(\langle w, x\rangle)\right) d\mu(x) \\
&= \alpha_1 \int_{[0,1]^d} g_1(\langle w, x\rangle)d\mu(x) + \alpha_2 \int_{[0,1]^d} g_2(\langle w, x\rangle)d\mu(x) \\
&= \alpha_1\psi(g_1) + \alpha_2\psi(g_2).
\end{aligned}$$

Furthermore, we have a continuity property of ψ of the form:

$$\begin{aligned}
|\psi(g_1) - \psi(g_2)| &= \left|\int_{[0,1]^d} (g_1 - g_2)(\langle w, x\rangle)d\mu(x)\right| \\
&= \left|\int_{[0,1]^d} (g_1 - g_2)(\langle w, x\rangle)h(x)d|\mu|(x)\right| \\
&\leq \int_{[0,1]^d} |(g_1 - g_2)(\langle w, x\rangle)|\,|h(x)|d|\mu|(x) \\
&\leq ||g_1 - g_2||_\infty \int_{[0,1]^d} |h(x)|d|\mu|(x),
\end{aligned}$$

with $||g_1 - g_2||_\infty = \sup_{t \in [-||w||_1, ||w||_1]} |g_1(t) - g_2(t)|$. Hence we have

$$|\psi(g_1) - \psi(g_2)| \leq ||g_1 - g_2||_\infty \int_{[0,1]^d} d|\mu|(x) = ||g_1 - g_2||_\infty \underbrace{|\mu|([0,1]^d)}_{<\infty},$$

which is a Lipschitz property (stronger than continuity). Then, for $\theta \in \mathbb{R}$ and $g : [-||w||_1, ||w||_1] \to \mathbb{R}$ defined by

$$g(t) = \mathbf{1}_{t \in [\theta, +\infty)}$$

for $t \in [-||w||_1, ||w||_1]$, we have

$$\begin{aligned}
\psi(g) &= \int_{[0,1]^d} \mathbf{1}_{\langle w,x \rangle \in [\theta, +\infty)} d\mu(x) \\
&= \int_{[0,1]^d} \mathbf{1}_{\langle w,x \rangle - \theta \geq 0} d\mu(x) \\
&= \int_{[0,1]^d} \mathbf{1}_{\langle w,x \rangle - \theta > 0} d\mu(x) + \int_{[0,1]^d} \mathbf{1}_{\langle w,x \rangle - \theta = 0} d\mu(x) \\
&= \int_{H_{w,-\theta}} d\mu(x) + \int_{\Pi_{w,-\theta}} d\mu(x) \\
&= \mu(H_{w,-\theta}) + \mu(\Pi_{w,-\theta}) \\
&= 0,
\end{aligned}$$

from what we have seen before. For g defined on $[-||w||_1, ||w||_1]$ valued in $\mathbb{C}$, defined by

$$g(t) = \mathbf{1}_{t \in (\theta, +\infty)}$$

for $t \in [-||w||_1, ||w||_1]$, we also have

$$\begin{aligned}
\psi(g) &= \int_{[0,1]^d} \mathbf{1}_{\langle w,x \rangle - \theta > 0} d\mu(x) \\
&= \mu(H_{w,-\theta}) \\
&= 0.
\end{aligned}$$

Hence,

- with

$$\mathbf{1}_{[\theta_1, \theta_2]} : [-||w||_1, ||w||_1] \to \mathbb{R}$$

defined by, for $t \in [-||w||_1, ||w||_1]$,

$$\mathbf{1}_{[\theta_1, \theta_2]}(t) = \mathbf{1}_{t \in [\theta_1, \theta_2]} = \mathbf{1}_{\theta_1 \leq t \leq \theta_2},$$

- with

$$\mathbf{1}_{[\theta_1, \theta_2)} : [-||w||_1, ||w||_1] \to \mathbb{R}$$

defined by, for $t \in [-||w||_1, ||w||_1]$,

$$\mathbf{1}_{[\theta_1, \theta_2)}(t) = \mathbf{1}_{t \in [\theta_1, \theta_2)} = \mathbf{1}_{\theta_1 \leq t < \theta_2},$$

- with
$$\mathbf{1}_{(\theta_1,\theta_2]} : [-||w||_1, ||w||_1] \to \mathbb{R}$$
defined by, for $t \in [-||w||_1, ||w||_1]$,
$$\mathbf{1}_{(\theta_1,\theta_2]}(t) = \mathbf{1}_{t\in(\theta_1,\theta_2]} = \mathbf{1}_{\theta_1<t\leq\theta_2},$$

we have
$$\psi(\mathbf{1}_{[\theta_1,\theta_2]}) = \psi(\mathbf{1}_{[\theta_1,+\infty)} - \mathbf{1}_{(\theta_2,+\infty)}),$$
with $\mathbf{1}_{[\theta_1,+\infty)}(t) = \mathbf{1}_{t\geq\theta_1}$ and $\mathbf{1}_{(\theta_2,+\infty)}(t) = \mathbf{1}_{t>\theta_2}$ (for $t \in [-||w||_1, ||w||_1]$). Hence
$$\psi(\mathbf{1}_{[\theta_1,\theta_2]}) = \psi(\mathbf{1}_{[\theta_1,+\infty)}) - \psi(\mathbf{1}_{(\theta_2,+\infty)}) = 0 - 0 = 0,$$
from what we have seen before. Also
$$\psi(\mathbf{1}_{[\theta_1,\theta_2)}) = \psi(\mathbf{1}_{[\theta_1,+\infty)}) - \psi(\mathbf{1}_{[\theta_2,+\infty)}) = 0 - 0 = 0$$
and
$$\psi(\mathbf{1}_{(\theta_1,\theta_2]}) = \psi(\mathbf{1}_{(\theta_1,+\infty)}) - \psi(\mathbf{1}_{(\theta_2,+\infty)}) = 0 - 0 = 0.$$
Now let us write $r : [-||w||_1, ||w||_1] \to \mathbb{C}$ defined by
$$r(t) = e^{it} = \cos(t) + i\sin(t),$$
with $i^2 = -1$ and for $t \in [-||w||_1, ||w||_1]$. Let us also write, for $k \in \mathbb{N}$ and $t \in [-||w||_1, ||w||_1]$,
$$r_k(t) = \mathbf{1}_{t=-||w||_1} r(-||w||_1) + \sum_{j=-k}^{k-1} \mathbf{1}_{(\frac{j||w||_1}{k}, \frac{(j+1)||w||_1}{k}]}(t) r\left(\frac{j||w||_1}{k}\right).$$
Then
$$\sup_{t\in[-||w||_1,||w||_1]} |r_k(t) - r(t)| \leq \sup_{\substack{x,y\in[-||w||_1,||w||_1] \\ |x-y|\leq\frac{||w||_1}{k}}} |r(x) - r(y)|$$
$$\underset{k\to\infty}{\to} 0,$$
since r is uniformly continuous (or even Lipschitz) on $[-||w||_1, ||w||_1]$. Hence, with the continuity property that we have seen,
$$\psi(r) = \lim_{k\to\infty} \psi(r_k) = 0,$$
since $\psi(r_k) = 0$ for $k \in \mathbb{N}$ from what we have seen before. Hence, we have shown that for any $w \in \mathbb{R}^d$,
$$\int_{[0,1]^d} e^{i\langle w,x\rangle} d\mu(x) = 0.$$
We see the Fourier transform of the measure μ. This implies that μ is the zero measure (which can be shown by technical arguments which are not specific to neural networks). Hence
$$L(f_0) = \int_{[0,1]^d} f_0(x) d\mu(x) = 0$$
which is a contradiction with $L(f_0) = 1$ and concludes the proof of Theorem 2.1.

There are two main take home messages.

- The density result $\overline{\mathcal{N}_1} = C([0,1]^d, \mathbb{R})$.
- The non-constructive proof technique, by contradiction. The use of the Hahn-Banach theorem to prove a density result is standard.

2.4 Other types of approximation results

2.4.1 About nonpolynomial activations

Theorem 2.5 (Hornik 1991). *Let $\sigma : \mathbb{R} \to \mathbb{R}$ be any bounded, measurable, nonpolynomial activation function. Then single-hidden-layer networks with activation σ are dense in $C([0,1]^d)$.*

Theorem 2.6 (Leshno, Lin, Pinkus, Schocken 1993). *Let $\sigma : \mathbb{R} \to \mathbb{R}$ be a locally bounded activation. Then single-layer networks with activation σ are dense in $C([0,1]^d)$ if and only if σ is not a polynomial almost everywhere.*

2.4.2 Constructive Universal Approximation with ReLU

Theorem 2.7 (Constructive ReLU Approximation). *Let $f \in C([0,1]^d)$ be L-Lipschitz continuous. Then for every $\varepsilon > 0$ there exists a ReLU network with $O(\varepsilon^{-d})$ units that satisfies*

$$\|f - g\|_\infty < \varepsilon.$$

Proof. Partition $[0,1]^d$ into cubes of side length h. On each cube, approximate f by its value at the lower-left vertex. Lipschitz continuity ensures the approximation error is bounded by Lh. Choose $h = \varepsilon/L$. A ReLU network can represent the resulting piecewise linear function exactly by encoding max-affine functions. □

CHAPTER 3 Generalization error and VC-dimension

The purpose of this first part is to properly introduce the notations and the notions of stochastic programming and how randomness can intervene in optimization methods.

3.1 Shattering coefficients

We consider a pair of random variables (X, Y) on $[0,1]^d \times \{0,1\}$. We consider $(X_1, Y_1), \dots, (X_n, Y_n)$ independent, with the same distribution as (X, Y) and independent of (X, Y). We consider a set $\mathcal{F}$ of functions from $[0,1]^d$ to $\{0,1\}$. Then, we have seen in Section 1.2 that the generalization error is

$$\mathbb{E}\left(\sup_{f\in\mathcal{F}}\left|\mathbb{P}\left(f(X)\neq Y\right)-\frac{1}{n}\sum_{i=1}^{n}\mathbf{1}_{f(X_i)\neq Y_i}\right|\right).$$

We have seen that, intuitively, the larger $\mathcal{F}$ is, the larger this generalization error is. A measure of the "size" or "complexity" of $\mathcal{F}$ is given by the following definition.

Definition 3.1. We call shattering coefficient of $\mathcal{F}$ (at n for $n \in \mathbb{N}$) the quantity

$$\Pi_{\mathcal{F}}(n) = \max_{x_1,\dots,x_n\in[0,1]^d} \operatorname{card}\left\{\left(f(x_1),\dots,f(x_n)\right); f\in\mathcal{F}\right\}.$$

We observe that $\Pi_{\mathcal{F}}(n)$ is increasing with respect to $\mathcal{F}$ and n,

- if $\mathcal{F}_1 \subset \mathcal{F}_2$, then $\Pi_{\mathcal{F}_1}(n) \leq \Pi_{\mathcal{F}_2}(n)$,
- $\Pi_{\mathcal{F}}(n) \leq \Pi_{\mathcal{F}}(n+1)$.

Remark 3.2. We always have

$$\Pi_{\mathcal{F}}(n) \leq \operatorname{card}\left\{(i_1,\dots,i_n)\in\{0,1\}^n\right\} = 2^n.$$

◀

Remark 3.3. If $\mathcal{F}$ is a finite set,

$$\Pi_{\mathcal{F}}(n) \leq \operatorname{card}(\mathcal{F}).$$

◀

Example
Let $d = 1$ and

$$\mathcal{F} = \{x \in [0,1] \mapsto \mathbf{1}_{x \geq a}; a \in \mathbb{R}\}.$$

Then for any $0 \leq x_1 \leq \cdots \leq x_n \leq 1$ and for $f \in \mathcal{F}$, we have

$$(f(x_1), \ldots, f(x_n)) = (0, \ldots, 0, 1, \ldots, 1),$$

where

- if $a > x_n$ then there are only 0s,
- if $a \leq x_1$ then here are only 1s,
- if $x_1 < a \leq x_n$ then the first 1 is at position $i \in \{2, \ldots, n\}$ with $x_{i-1} < a$ and $x_i \geq a$.

Hence the vectors that we can obtain are

$$(0, \ldots, 0), (0, \ldots, 1), (0, \ldots, 1, 1), \ldots, (1, \ldots, 1).$$

Hence there are $n+1$ possibilities. Hence

$$\operatorname{card}\{(f(x_1), \ldots, f(x_n)); f \in \mathcal{F}\} \leq n+1.$$

If we consider $x_1, \ldots, x_n$ that are not necessarily ordered, there is a bijection between $\{(f(x_1), \ldots, f(x_n)); f \in \mathcal{F}\}$ and $\left\{\left(f(x_{(1)}), \ldots, f(x_{(n)})\right); f \in \mathcal{F}\right\}$ where $x_{(1)} \leq \cdots \leq x_{(n)}$ are obtained by ordering $x_1, \ldots, x_n$. Thus we still have

$$\operatorname{card}\{(f(x_1), \ldots, f(x_n)); f \in \mathcal{F}\} \leq n+1.$$

Hence $\Pi_{\mathcal{F}}(n) \leq n+1$. Furthermore, with $x_1 = 0, x_2 = 1/n, \ldots, x_n = (n-1)/n$,

- with f given by $x \mapsto \mathbf{1}_{x \geq 2}$ we have $(f(x_1), \ldots, f(x_n)) = (0, \ldots, 0)$,
- with f given by $x \mapsto \mathbf{1}_{x \geq -1}$ we have $(f(x_1), \ldots, f(x_n)) = (1, \ldots, 1)$,
- for $i \in \{1, \ldots, n-1\}$, with f given by $x \mapsto \mathbf{1}_{x \geq (x_i + x_{i+1})/2}$ we have $(f(x_1), \ldots, f(x_n)) = (0, \ldots, 0, 1, \ldots, 1)$ with i 0s and $n-i$ 1s.

Hence with $x_1 = 0, x_2 = 1/n, \ldots, x_n = (n-1)/n$ we have

$$\operatorname{card}\{(f(x_1), \ldots, f(x_n)); f \in \mathcal{F}\} \geq n+1.$$

Hence finally $\Pi_{\mathcal{F}}(n) \geq n+1$ and thus

$$\Pi_{\mathcal{F}}(n) = n+1.$$

Example
Let $d = 2$ and

$$\mathcal{F} = \left\{x \in [0,1]^2 \mapsto \mathbf{1}_{\langle w, x \rangle \geq a}; a \in \mathbb{R}; w \in \mathbb{R}^2\right\}.$$

These are affine classifiers as in Figure 3.1.

Then for $n = 3$ and for $x_1, x_2, x_3 \in [0,1]^2$ that are not contained in a line, we can obtain the 8 possible classification vectors, as shown in Figure 3.2.

Hence

$$\operatorname{card}\{(f(x_1), f(x_2), f(x_3)); f \in \mathcal{F}\} \geq 8.$$

Also

$$\operatorname{card}\{(f(x_1), f(x_2), f(x_3)); f \in \mathcal{F}\} \leq \operatorname{card}\left\{(i_1, i_2, i_3) \in \{0,1\}^3\right\} = 2^3 = 8.$$

Hence we have

$$\Pi_{\mathcal{F}}(3) = 8.$$

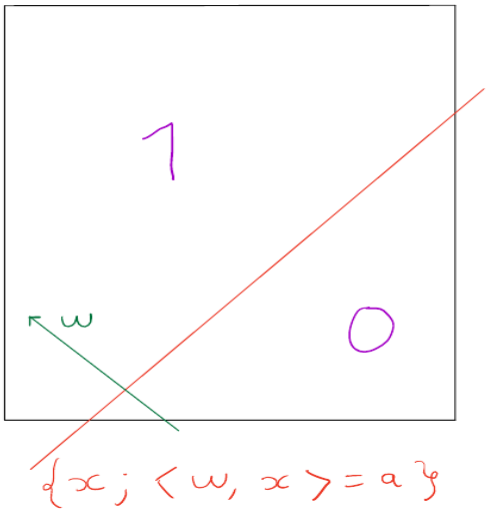

Figure 3.1: An example of an affine classifier.

3.2 Bounding the generalization error from the shattering coefficients

The next proposition enables to bound the generalization error from $\Pi_{\mathcal{F}}(n)$.

Proposition 3.4. *For any set $\mathcal{F}$ of functions from $[0,1]^d$ to $\{0,1\}$, we have*

$$\mathbb{E}\left(\sup_{f\in\mathcal{F}}\left|\mathbb{P}\left(f(X)\neq Y\right)-\frac{1}{n}\sum_{i=1}^{n}\mathbf{1}_{f(X_i)\neq Y_i}\right|\right)\leq 2\sqrt{\frac{2\log(2\Pi_{\mathcal{F}(n)})}{n}}.$$

Remarks

- The notation log stands for the Neper base e logarithm.
- We see a dependence in $1/\sqrt{n}$, which is classical when empirical means are compared with expectations.

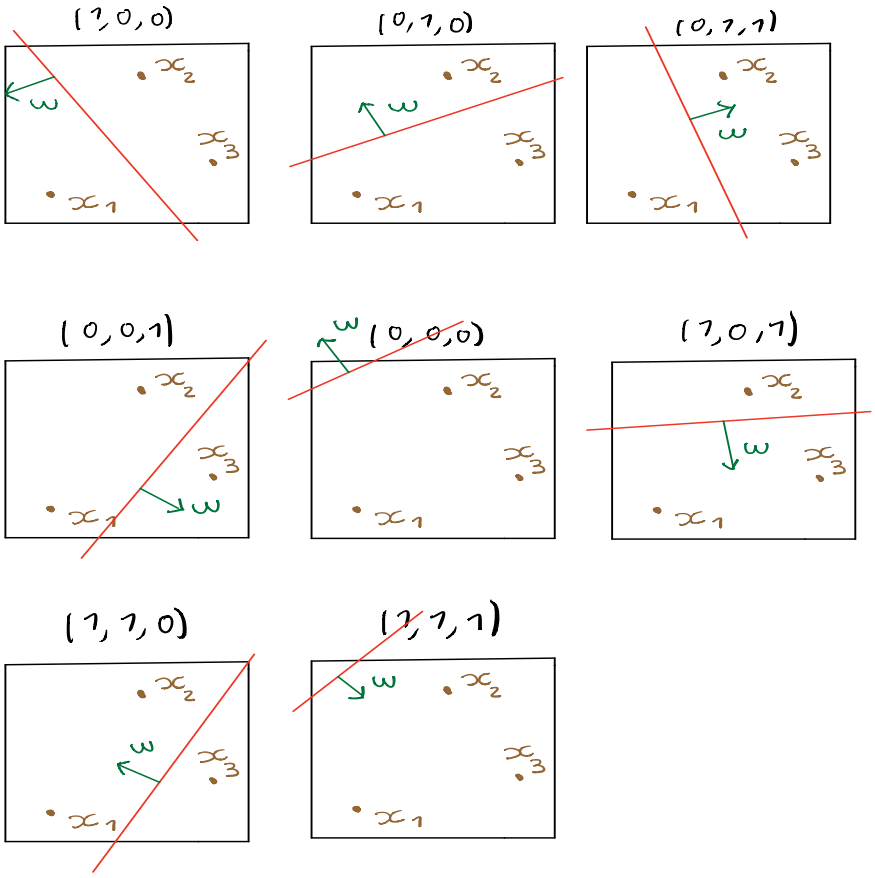

Figure 3.2: Obtaining the 8 possible classification vectors with 3 points and affine classifiers.

- If $\text{card}(\mathcal{F}) = 1$ with $\mathcal{F} = \{f\}$ then

$$\begin{aligned}
&\mathbb{E}\left(\sup_{f\in\mathcal{F}}\left|\mathbb{P}\left(f(X)\neq Y\right) - \frac{1}{n}\sum_{i=1}^{n}\mathbf{1}_{f(X_i)\neq Y_i}\right|\right)\\
&= \mathbb{E}\left(\left|\mathbb{E}\left(\mathbf{1}_{f(X)\neq Y}\right) - \frac{1}{n}\sum_{i=1}^{n}\mathbf{1}_{f(X_i)\neq Y_i}\right|\right)\\
&\leq \sqrt{\mathbb{E}\left(\left(\mathbb{E}\left(\mathbf{1}_{f(X)\neq Y}\right) - \frac{1}{n}\sum_{i=1}^{n}\mathbf{1}_{f(X_i)\neq Y_i}\right)^2\right)}\\
&= \sqrt{\text{Var}\left(\frac{1}{n}\sum_{i=1}^{n}\mathbf{1}_{f(X_i)\neq Y_i}\right)}\\
&= \sqrt{\frac{1}{n^2}\text{Var}\left(\sum_{i=1}^{n}\mathbf{1}_{f(X_i)\neq Y_i}\right)}\\
&= \sqrt{\frac{1}{n}}
\end{aligned}$$

In the second inequality above, we have used Jensen's inequality which implies that $\mathbb{E}(|W|) \leq \sqrt{\mathrm{Var}(W)}$ for a random variable W. In the second equality above we have used that $(X_1, Y_1), \ldots, (X_n, Y_n)$ are independent and distributed as (X, Y). The first inequality above holds because

$$\mathrm{Var}\left(\mathbf{1}_{f(X)\neq Y}\right) \leq \mathbb{E}\left(\left(\mathbf{1}_{f(X)\neq Y}\right)^2\right) \leq \mathbb{E}(1) = 1.$$

On the other hand the upper bound of Proposition 3.4 is

$$2\sqrt{\frac{2\log(2)}{n}} = \underbrace{2\sqrt{2\log(2)}}_{\approx 2.35}\frac{1}{\sqrt{n}}.$$

We obtain the same order of magnitude $1/\sqrt{n}$.

- In all cases, we have $\Pi_{\mathcal{F}}(n) \leq 2^n$ and thus the bound of Proposition 3.4 is smaller than

$$2\sqrt{\frac{2\log(2\times 2^n)}{n}} = 2\sqrt{\frac{2\log(2^{n+1})}{n}} = 2\sqrt{\frac{2(n+1)\log(2)}{n}} = 2\sqrt{2\log(2)\left(1+\frac{1}{n}\right)} \underset{n\to\infty}{\to} 2\sqrt{2\log(2)}.$$

This bound based on $\Pi_{\mathcal{F}}(n) \leq 2^n$ is not informative because we already know that

$$\mathbb{E}\left(\sup_{f\in\mathcal{F}}\left|\underbrace{\mathbb{P}\left(f(X)\neq Y\right)}_{\in[0,1]} - \underbrace{\frac{1}{n}\sum_{i=1}^{n}\mathbf{1}_{f(X_i)\neq Y_i}}_{\in[0,1]}\right|\right) \leq \mathbb{E}\left(\sup_{f\in\mathcal{F}} 1\right) = 1.$$

To summarize

- The bound of Proposition 3.4 agrees in terms of order of magnitudes with the two extreme cases $\mathrm{card}(\mathcal{F}) = 1$ (then $\Pi_{\mathcal{F}}(n) = 1$) and $\Pi_{\mathcal{F}}(n) \leq 2^n$.
- This bound will be particularly useful when $\Pi_{\mathcal{F}}(n)$ is in between these two cases.

Proof of Proposition 3.4

The proof is based on a classical argument that is called symmetrization. Without loss of generality, we can assume that $Y \in \{-1, 1\}$ and $\mathcal{F}$ is composed of functions from $[0,1]^d$ to $\{-1,1\}$ (the choice of 0 and 1 to define the two classes is arbitrary in classification, here -1 and 1 will be more convenient). We let $(\tilde{X}_1, \tilde{Y}_1), \ldots, (\tilde{X}_n, \tilde{Y}_n)$ be pairs of random variables such that $(X_1, Y_1), \ldots, (X_n, Y_n), (\tilde{X}_1, \tilde{Y}_1), \ldots, (\tilde{X}_n, \tilde{Y}_n)$ are independent and with the same distribution as (X, Y).

Then

$$\mathbb{P}\left(f(X)\neq Y\right) = \tilde{\mathbb{E}}\left(\frac{1}{n}\sum_{i=1}^{n}\mathbf{1}_{f(\tilde{X}_i)\neq\tilde{Y}_i}\right),$$

writing $\tilde{\mathbb{E}}$ to indicate that the expectation is taken with respect to $(\tilde{X}_1, \tilde{Y}_1), \ldots, (\tilde{X}_n, \tilde{Y}_n)$. We let

$$\Delta_n = \mathbb{E}\left(\sup_{f\in\mathcal{F}}\left|\mathbb{P}\left(f(X)\neq Y\right) - \frac{1}{n}\sum_{i=1}^{n}\mathbf{1}_{f(X_i)\neq Y_i}\right|\right).$$

We have

$$\begin{aligned}\Delta_n &= \mathbb{E}\left(\sup_{f\in\mathcal{F}}\left|\tilde{\mathbb{E}}\left(\frac{1}{n}\sum_{i=1}^n \mathbf{1}_{f(\tilde{X}_i)\neq\tilde{Y}_i}\right) - \frac{1}{n}\sum_{i=1}^n \mathbf{1}_{f(X_i)\neq Y_i}\right|\right)\\ &= \mathbb{E}\left(\sup_{f\in\mathcal{F}}\left|\tilde{\mathbb{E}}\left(\frac{1}{n}\sum_{i=1}^n \mathbf{1}_{f(\tilde{X}_i)\neq\tilde{Y}_i} - \frac{1}{n}\sum_{i=1}^n \mathbf{1}_{f(X_i)\neq Y_i}\right)\right|\right)\\ &\leq \mathbb{E}\left(\sup_{f\in\mathcal{F}}\tilde{\mathbb{E}}\left(\left|\frac{1}{n}\sum_{i=1}^n \mathbf{1}_{f(\tilde{X}_i)\neq\tilde{Y}_i} - \frac{1}{n}\sum_{i=1}^n \mathbf{1}_{f(X_i)\neq Y_i}\right|\right)\right)\\ &\leq \mathbb{E}\tilde{\mathbb{E}}\left(\sup_{f\in\mathcal{F}}\left|\frac{1}{n}\sum_{i=1}^n \mathbf{1}_{f(\tilde{X}_i)\neq\tilde{Y}_i} - \frac{1}{n}\sum_{i=1}^n \mathbf{1}_{f(X_i)\neq Y_i}\right|\right).\end{aligned}$$

Let now $\sigma_1,\dots,\sigma_n$ be independent random variables, independent from $(X_i, Y_i, \tilde{X}_i, \tilde{Y}_i)_{i=1,\dots,n}$ and such that

$$\mathbb{P}_\sigma(\sigma_i = 1) = \mathbb{P}_\sigma(\sigma_i = -1) = \frac{1}{2}$$

for $i = 1,\dots,n$ and by writing $\mathbb{E}_\sigma$ and $\mathbb{P}_\sigma$ the expectations and probabilities with respect to $\sigma_1,\dots,\sigma_n$. We let for $i = 1,\dots,n$

$$(\bar{X}_i, \bar{Y}_i) = \begin{cases}(X_i, Y_i) & \text{if } \sigma_i = 1\\ (\tilde{X}_i, \tilde{Y}_i) & \text{if } \sigma_i = -1\end{cases}$$

and

$$(\bar{\tilde{X}}_i, \bar{\tilde{Y}}_i) = \begin{cases}(\tilde{X}_i, \tilde{Y}_i) & \text{if } \sigma_i = 1\\ (X_i, Y_i) & \text{if } \sigma_i = -1\end{cases}.$$

Then $(\bar{X}_i, \bar{Y}_i)_{i=1,\dots,n}, (\bar{\tilde{X}}_i, \bar{\tilde{Y}}_i)_{i=1,\dots,n}$ are independent and have the same distribution as (X, Y). Let us show this. For any bounded measurable functions $g_1,\dots,g_{2n}$ from $[0,1]^d \times \{-1,1\}$ to $\mathbb{R}$, we have, using the law of total expectation,

$$\mathbb{E}\left(\left(\prod_{i=1}^n g_i(\bar{X}_i, \bar{Y}_i)\right)\left(\prod_{i=1}^n g_{n+i}(\bar{\tilde{X}}_i, \bar{\tilde{Y}}_i)\right)\right) = \mathbb{E}\left(\mathbb{E}\left(\left(\prod_{i=1}^n g_i(\bar{X}_i, \bar{Y}_i)\right)\left(\prod_{i=1}^n g_{n+i}(\bar{\tilde{X}}_i, \bar{\tilde{Y}}_i)\right)\middle|\sigma_1,\dots,\sigma_n\right)\right)$$

$$= \mathbb{E}\left(\mathbb{E}\left(\left(\prod_{\substack{i=1\\ \sigma_i=1}}^n g_i(X_i, Y_i)\right)\left(\prod_{\substack{i=1\\ \sigma_i=-1}}^n g_i(\tilde{X}_i, \tilde{Y}_i)\right)\left(\prod_{\substack{j=1\\ \sigma_j=1}}^n g_{n+j}(\tilde{X}_j, \tilde{Y}_j)\right)\left(\prod_{\substack{j=1\\ \sigma_j=-1}}^n g_{n+j}(X_j, Y_j)\right)\middle|\sigma_1,\dots,\sigma_n\right)\right).$$

In the above conditional expectation, the $2n$ variables are independent since each of the $(X_i, Y_i)_{i=1,\dots,n}, (X_i, Y_i)_{i=1,\dots,n}$ appears exactly once. Their common distribution is that of (X, Y). Furthermore, the $2n$ functions $g_1,\dots,g_{2n}$ appear once each. Hence we have

$$\mathbb{E}\left(\left(\prod_{i=1}^n g_i(\bar{X}_i, \bar{Y}_i)\right)\left(\prod_{i=1}^n g_{n+i}(\bar{\tilde{X}}_i, \bar{\tilde{Y}}_i)\right)\right) = \mathbb{E}_\sigma\left(\prod_{i=1}^{2n}\mathbb{E}\left(g_i(X, Y)\right)\middle|\sigma_1,\dots,\sigma_n\right) = \prod_{i=1}^{2n}\mathbb{E}\left(g_i(X, Y)\right).$$

Hence, indeed, $(\bar{X}_i, \bar{Y}_i)_{i=1,\dots,n}, (\bar{\tilde{X}}_i, \bar{\tilde{Y}}_i)_{i=1,\dots,n}$ are independent and have the same distribution as (X, Y).

Hence, we have

$$\Delta_n \leq \mathbb{E}\tilde{\mathbb{E}}\mathbb{E}_\sigma\left(\sup_{f\in\mathcal{F}}\left|\frac{1}{n}\sum_{i=1}^n\left(\mathbf{1}_{f(\bar{X}_i)\neq\bar{Y}_i}-\mathbf{1}_{f(\bar{\bar{X}}_i)\neq\bar{\bar{Y}}_i}\right)\right|\right),$$

where $\mathbb{E}_\sigma$ means that only $\sigma_1,\dots\sigma_n$ are random. We observe that

$$\left(\mathbf{1}_{f(\bar{X}_i)\neq\bar{Y}_i}-\mathbf{1}_{(\bar{\bar{X}}_i)\neq\bar{\bar{Y}}_i}\right)=\sigma_i\left(\mathbf{1}_{f(X_i)\neq Y_i}-\mathbf{1}_{f(\tilde{X}_i)\neq\tilde{Y}_i}\right)$$

because

- if $\sigma_i=1$, $(\bar{X}_i,\bar{Y}_i,\bar{\bar{X}}_i,\bar{\bar{Y}}_i)=(X_i,Y_i,\tilde{X}_i,\tilde{Y}_i)$,
- if $\sigma_i=-1$, $(\bar{X}_i,\bar{Y}_i,\bar{\bar{X}}_i,\bar{\bar{Y}}_i)=(\tilde{X}_i,\tilde{Y}_i,X_i,Y_i)$.

Hence

$$\begin{aligned}
\Delta_n &\leq \mathbb{E}\tilde{\mathbb{E}}\mathbb{E}_\sigma\left(\sup_{f\in\mathcal{F}}\left|\frac{1}{n}\sum_{i=1}^n\sigma_i\left(\mathbf{1}_{f(X_i)\neq Y_i}-\mathbf{1}_{f(\tilde{X}_i)\neq\tilde{Y}_i}\right)\right|\right)\\
&\leq \mathbb{E}\tilde{\mathbb{E}}\mathbb{E}_\sigma\left(\sup_{f\in\mathcal{F}}\left|\frac{1}{n}\sum_{i=1}^n\sigma_i\mathbf{1}_{f(X_i)\neq Y_i}\right|+\sup_{f\in\mathcal{F}}\left|\frac{1}{n}\sum_{i=1}^n\sigma_i\mathbf{1}_{f(\tilde{X}_i)\neq\tilde{Y}_i}\right|\right)\\
&= 2\mathbb{E}\mathbb{E}_\sigma\left(\sup_{f\in\mathcal{F}}\left|\frac{1}{n}\sum_{i=1}^n\sigma_i\mathbf{1}_{f(X_i)\neq Y_i}\right|\right)\\
&\leq 2\max_{y_1,\dots,y_n\in\{-1,1\}}\max_{x_1,\dots,x_n\in[0,1]^d}\mathbb{E}_\sigma\left(\sup_{f\in\mathcal{F}}\left|\frac{1}{n}\sum_{i=1}^n\sigma_i\mathbf{1}_{f(x_i)\neq y_i}\right|\right).
\end{aligned}$$

For any $y_1,\dots,y_n\in\{-1,1\}$ and $x_1,\dots,x_n\in[0,1]^d$ we define the set

$$V_{\mathcal{F}}(x,y)=\left\{\left(\mathbf{1}_{y_1\neq f(x_1)},\dots,\mathbf{1}_{y_n\neq f(x_n)}\right);f\in\mathcal{F}\right\}.$$

Then we have

$$\Delta_n\leq\frac{2}{n}\max_{y_1,\dots,y_n\in\{-1,1\}}\max_{x_1,\dots,x_n\in[0,1]^d}\mathbb{E}_\sigma\left(\sup_{v\in V_{\mathcal{F}}(x,y)}|\langle\sigma,v\rangle|\right),$$

with $\sigma=(\sigma_1,\dots,\sigma_n)$.

We observe that for all $x_1,\dots,x_n,y_1,\dots,y_n$, there is a bijection between $V_{\mathcal{F}}(x,y)$ and $\{(f(x_1),\dots,f(x_n));f\in\mathcal{F}\}$. Hence

$$\max_{y_1,\dots,y_n\in\{-1,1\}}\max_{x_1,\dots,x_n\in[0,1]^d}\operatorname{card}(V_{\mathcal{F}}(x,y))\leq\Pi_{\mathcal{F}}(n).$$

Assume that we show

$$\text{For any set } V\subset\{-1,0,1\}^n,\ \mathbb{E}_\sigma\left(\sup_{v\in V}|\langle\sigma,v\rangle|\right)\leq\sqrt{2n\log(2\operatorname{card}(V))}. \tag{3.1}$$

Then we would have

$$\Delta_n\leq\frac{2}{n}\sqrt{2n\log(2\Pi_{\mathcal{F}}(n))}=2\sqrt{\frac{2}{n}\log(2\Pi_{\mathcal{F}}(n))}$$

which would conclude the proof.

Let us now show (3.1). Let us write $-V = \{-v; v \in V\}$ and $V^{\#} = V \cup -V$. We have, for any $s > 0$,

$$\mathbb{E}_\sigma\left(\sup_{v\in V}|\langle\sigma,v\rangle|\right) = \mathbb{E}_\sigma\left(\sup_{v\in V^{\#}}\langle\sigma,v\rangle\right) = \mathbb{E}_\sigma\left(\frac{1}{s}\log\left(e^{s\sup_{v\in V^{\#}}\langle\sigma,v\rangle}\right)\right).$$

We now apply Jensen inequality to the concave function $(1/s)\log$. This gives

$$\begin{aligned}
\mathbb{E}_\sigma\left(\sup_{v\in V}|\langle\sigma,v\rangle|\right) &\le \frac{1}{s}\log\left(\mathbb{E}_\sigma\left(e^{s\sup_{v\in V^{\#}}\langle\sigma,v\rangle}\right)\right)\\
&= \frac{1}{s}\log\left(\mathbb{E}_\sigma\left(\sup_{v\in V^{\#}} e^{s\langle\sigma,v\rangle}\right)\right)\\
&\le \frac{1}{s}\log\left(\mathbb{E}_\sigma\left(\sum_{v\in V^{\#}} e^{s\langle\sigma,v\rangle}\right)\right)\\
&= \frac{1}{s}\log\left(\sum_{v\in V^{\#}}\mathbb{E}_\sigma\left(e^{s\langle\sigma,v\rangle}\right)\right)\\
&= \frac{1}{s}\log\left(\sum_{v\in V^{\#}}\prod_{i=1}^{n}\mathbb{E}_\sigma\left(e^{s\sigma_i v_i}\right)\right)\\
&= \frac{1}{s}\log\left(\sum_{v\in V^{\#}}\prod_{i=1}^{n}\frac{1}{2}\left(e^{sv_i}+e^{-sv_i}\right)\right).
\end{aligned}$$

We can show simply that for $x \ge 0$, $e^x + e^{-x} \le 2e^{x^2/2}$. This gives, using also that $v_i^2 \le 1$ for $i = 1, \dots, n$ and $v \in V$,

$$\begin{aligned}
\mathbb{E}_\sigma\left(\sup_{v\in V}|\langle\sigma,v\rangle|\right) &\le \frac{1}{s}\log\left(\sum_{v\in V^{\#}}\prod_{i=1}^{n}e^{\frac{s^2v_i^2}{2}}\right)\\
&\le \frac{1}{s}\log\left(\sum_{v\in V^{\#}}e^{\frac{ns^2}{2}}\right)\\
&\le \frac{1}{s}\log\left(\mathrm{card}(V^{\#})e^{\frac{ns^2}{2}}\right)\\
&= \frac{\log(\mathrm{card}(V^{\#}))}{s} + \frac{ns}{2}.
\end{aligned}$$

We let

$$s = \sqrt{\frac{2\log(\mathrm{card}(V^{\#}))}{n}}$$

which gives

$$\begin{aligned}
&\mathbb{E}_\sigma\left(\sup_{v\in V}|\langle\sigma,v\rangle|\right) \le \frac{1}{\sqrt{2}}\sqrt{n\log(\mathrm{card}(V^{\#}))} + \frac{1}{\sqrt{2}}\sqrt{n\log(\mathrm{card}(V^{\#}))} = 2\sqrt{\frac{n}{2}\log(\mathrm{card}(V^{\#}))}\\
&= \sqrt{2n\log(\mathrm{card}(V^{\#}))} \le \sqrt{2n\log(2\mathrm{card}(V))}.
\end{aligned}$$

Hence (3.1) is proved and thus the proof of Proposition 3.4 is concluded. □

3.3 VC-dimension

From the previous proposition, the shattering coefficient $\Pi_{\mathcal{F}}(n)$ is important and we would like to quantify its growth as n grows. A tool for this is the Vapnik-Cherbonenkis dimension, that we will call VC-dimension.

Definition 3.5. For a set $\mathcal{F}$ of functions from $[0,1]^d$ to $\{0,1\}$, we write $\mathrm{VCdim}(\mathcal{F})$ and call VC-dimension the quantity

$$\mathrm{VCdim}(\mathcal{F}) = \sup\{m \in \mathbb{N}; \Pi_{\mathcal{F}}(m) = 2^m\}$$

with the convention $\Pi_{\mathcal{F}}(0) = 1$ so that $\mathrm{VCdim}(\mathcal{F}) \geq 0$. It is possible that $\mathrm{VCdim}(\mathcal{F}) = +\infty$.

Interpretation

The quantity $\mathrm{VCdim}(\mathcal{F})$ is the largest number of input points that can be "shattered", meaning that they can be classified in all possible ways by varying the classifier in $\mathcal{F}$.

Examples

- When
$$\mathcal{F} = \{\text{all the functions from } [0,1]^d \text{ to } \{0,1\}\}$$
then $\mathrm{VCdim}(\mathcal{F}) = +\infty$. Indeed, for any $n \in \mathbb{N}$, by considering $x_1, \ldots, x_n$ two-by-two distinct, we have $\Pi_{\mathcal{F}}(n) = 2^n$.
- When $\mathcal{F}$ is finite with $\mathrm{card}(\mathcal{F}) \leq 2^{m_0}$ then $\mathrm{VCdim}(\mathcal{F}) \leq m_0$. Indeed, for $m > m_0$, we have seen that $\Pi_{\mathcal{F}}(m) \leq \mathrm{card}(\mathcal{F}) \leq 2^{m_0} \leq 2^m$. Hence
$$m \notin \{m \in \mathbb{N}; \Pi_{\mathcal{F}}(m) = 2^m\}$$
and thus
$$\sup\{m \in \mathbb{N}; \Pi_{\mathcal{F}}(m) = 2^m\} \leq m_0.$$

Remark 3.6. If $\mathrm{VCdim}(\mathcal{F}) = V < \infty$ then for $i = 1, \ldots, V$, $\Pi_{\mathcal{F}}(i) = 2^i$. Furthermore, if $\mathrm{VCdim}(\mathcal{F}) = \infty$ then for $i \in \mathbb{N}$, $\Pi_{\mathcal{F}}(i) = 2^i$. ◀

Proof of Remark 3.6

Let us start by the case $\mathrm{VCdim}(\mathcal{F}) = V < \infty$. Since $\Pi_{\mathcal{F}}(V) = 2^V$, there exist $x_1, \ldots, x_V \in [0,1]^d$ such that

$$\mathrm{card}\{(f(x_1), \ldots, f(x_V)); f \in \mathcal{F}\} = 2^V.$$

This means that we obtain all the possible vectors with components in $\{0,1\}$ and thus we obtain all the possible subvectors for the i first coefficients for $i = 1, \ldots, V$. Hence

$$\mathrm{card}\{(f(x_1), \ldots, f(x_i)); f \in \mathcal{F}\} = 2^i.$$

and thus $\Pi_{\mathcal{F}}(i) = 2^i$.

The proof for the case $\mathrm{VCdim}(\mathcal{F}) = \infty$ is similar. □

Similarly if for $i_0 \in \mathbb{N}$, $\Pi_{\mathcal{F}}(i_0) = 2^{i_0}$ then for all $i = 1, \ldots, i_0$, $\Pi_{\mathcal{F}}(i) = 2^i$.

We can compute the VC-dimension in the case of linear and affine classifiers.

Proposition 3.7. *Let $d \in \mathbb{N}$. Let*

$$\mathcal{F}_{d,l} = \left\{x \in [0,1]^d \mapsto \mathbf{1}_{\langle w,x\rangle \geq 0}; w \in \mathbb{R}^d\right\}$$

and

$$\mathcal{F}_{d,a} = \left\{ x \in [0,1]^d \mapsto \mathbf{1}_{\langle w,x\rangle + a \geq 0}; w \in \mathbb{R}^d, a \in \mathbb{R} \right\}.$$

Then

$$\mathrm{VCdim}(\mathcal{F}_{d,l}) = d$$

and

$$\mathrm{VCdim}(\mathcal{F}_{d,a}) = d + 1.$$

Remark 3.8. The VC-dimension coincides here with the number of free parameters and thus with the usual notion of dimension. ◀

Proof of Proposition 3.7 Write

$$x_1 = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \; x_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \; \ldots, \; x_d = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}$$

in $\mathbb{R}^d$. Then for any $y_1, \ldots, y_d \in \{0,1\}$ write

$$z_i = \begin{cases} 1 & \text{if } y_i = 1 \\ -1 & \text{if } y_i = 0 \end{cases}.$$

Consider

$$x \mapsto \mathbf{1}_{\langle x, \sum_{j=1}^d z_j x_j \rangle \geq 0}.$$

Then for $k = 1, \ldots, d$,

$$\mathbf{1}_{\langle x_k, \sum_{j=1}^d z_j x_j \rangle \geq 0} = \mathbf{1}_{\langle x_k, z_k x_k \rangle \geq 0} = \mathbf{1}_{z_k \geq 0} = y_k.$$

Hence we reach all the elements of $\{0,1\}^d$. Hence

$$\Pi_{\mathcal{F}_{d,l}}(d) = 2^d$$

and thus

$$\mathrm{VCdim}(\mathcal{F}_{d,l}) \geq d.$$

Assume that

$$\mathrm{VCdim}(\mathcal{F}_{d,l}) \geq d + 1.$$

Then, from Remark 3.6, $\Pi_{\mathcal{F}_{d,l}}(d+1) = 2^{d+1}$. Hence, there exists $x_1, \ldots, x_{d+1} \in [0,1]^d$ and $w_1, \ldots, w_{2^{d+1}} \in \mathbb{R}^d$ such that

$$\begin{pmatrix} w_i^\top x_1 \\ \vdots \\ w_i^\top x_{d+1} \end{pmatrix},$$

for $i = 1, \ldots, 2^{d+1}$ take all possible sign vectors (< 0 or ≥ 0). We write

$$X = \begin{pmatrix} x_1^\top \\ \vdots \\ x_{d+1}^\top \end{pmatrix}$$

of dimension $(d+1) \times d$ and

$$W = \begin{pmatrix} w_1 \dots w_{2^{d+1}} \end{pmatrix}$$

of dimension $d \times 2^{d+1}$. Then

$$XW = \begin{pmatrix} x_1^\top w_1 & \dots & x_1^\top w_{2^{d+1}} \\ \vdots & \dots & \vdots \\ x_{d+1}^\top w_1 & \dots & x_{d+1}^\top w_{2^{d+1}} \end{pmatrix}$$

is of dimension $(d+1) \times 2^{d+1}$. Let us show that the $d+1$ rows of XW are linearly independent. Let a of size $(d+1) \times 1$, non-zero such that

$$a^\top XW = 0,$$

where the above display is a linear combination of the rows of XW. Then, for $k \in \{1, \dots, 2^{d+1}\}$,

$$(a^\top XW)_k = a^\top X w_k = a^\top \begin{pmatrix} x_1^\top w_k \\ \vdots \\ x_{d+1}^\top w_k \end{pmatrix}.$$

Let k such that for $i = 1, \dots, d+1$, $a_i \geq 0$ if and only if $x_i^\top w_k \geq 0$ (k exists since we reach all the possible sign vectors). Then

$$(a^\top XW)_k = \sum_{i=1}^{d+1} \underbrace{a_i (x_i^\top w_k)}_{\text{same signs}} = \sum_{i=1}^{d+1} |a_i| |x_i^\top w_k|.$$

Since a is non-zero we can assume that there is a j such that $a_j < 0$ (up to replacing a by $-a$ at the beginning). Then

$$(a^\top XW)_k \geq |a_j| |x_j^\top w_k| > 0,$$

since $x_j^\top w_k < 0$ and $a_j < 0$. This is a contradiction. Hence there does no exist a of size $(d+1) \times 1$, non-zero such that $a^\top XW = 0$. Hence the $d+1$ lines of XW are linearly independent. Hence the rank of XW is equal to $d+1$. But the rank of XW is smaller or equal to d because X is of dimension $(d+1) \times d$. Hence we have reached a contradiction and thus

$$\text{VCdim}(\mathcal{F}_{d,l}) < d+1.$$

Hence

$$\text{VCdim}(\mathcal{F}_{d,l}) = d.$$

Let us now consider $\mathcal{F}_{d,a}$. Let

$$x_1 = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \; x_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \; \dots, \; x_d = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}$$

and

$$x_{d+1} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix},$$

in $\mathbb{R}^d$. Then, for any $y_1, \dots, y_{d+1} \in \{0,1\}$, write for $i = 1, \dots, d+1$,

$$z_i = \begin{cases} 1 & \text{if } y_i = 1 \\ -1 & \text{if } y_i = 0. \end{cases}.$$

Consider the function

$$x \in [0,1]^d \mapsto \mathbf{1}_{\langle x, \sum_{j=1}^d (z_j - z_{d+1}) x_j \rangle \geq -z_{d+1}}.$$

Then for $k = 1, \dots, d$,

$$\mathbf{1}_{\langle x_k, \sum_{j=1}^d (z_j - z_{d+1}) x_j \rangle \geq -z_{d+1}} = \mathbf{1}_{\langle x_k, (z_k - z_{d+1}) x_k \rangle \geq -z_{d+1}} = \mathbf{1}_{z_k - z_{d+1} \geq -z_{d+1}} = \mathbf{1}_{z_k \geq 0} = y_k.$$

and

$$\mathbf{1}_{\langle x_{d+1}, \sum_{j=1}^d (z_j - z_{d+1}) x_j \rangle \geq -z_{d+1}} = \mathbf{1}_{0 \geq -z_{d+1}} = \mathbf{1}_{z_{d+1} \geq 0} = y_{d+1}.$$

Hence we reach the 2^{d+1} possible vectors and thus

$$\mathrm{VCdim}(\mathcal{F}_{d,a}) \geq d + 1.$$

Assume now that

$$\mathrm{VCdim}(\mathcal{F}_{d,a}) \geq d + 2.$$

Then, as seen previously,

$$\Pi_{\mathcal{F}_{d,a}}(d+2) = 2^{d+2}.$$

Hence there exists $x_1, \dots, x_{d+2} \in [0,1]^d$ such that for all $y_1, \dots, y_{d+2} \in \{0,1\}$, there exists $w \in \mathbb{R}^d$ and $b \in \mathbb{R}$ such that, for $k = 1, \dots, d+2$,

$$\mathbf{1}_{\langle w, x_k \rangle + b \geq 0} = y_k.$$

We write

$$\bar{x_i} = \begin{pmatrix} x_i \\ 1 \end{pmatrix}$$

of size $(d+1) \times 1$ for $i = 1, \dots, d+2$ and

$$\bar{w} = \begin{pmatrix} w \\ b \end{pmatrix}$$

of size $(d+1) \times 1$. Then, for $k = 1, \dots, d+2$,

$$1_{\langle \bar{w}, \bar{x}_k \rangle \geq 0} = 1_{\langle w, x_k \rangle + b \geq 0} = y_k.$$

Hence in $\mathbb{R}^{d+1}$ we have shattered $d+2$ vectors $\bar{x}_1, \dots, \bar{x}_{d+2}$ (we have obtained all the possible sign vectors) with linear classifiers. This implies

$$\mathrm{VCdim}(\mathcal{F}_{d+1,l}) \geq d + 2$$

which is false since we have shown above that $\mathrm{VCdim}(\mathcal{F}_{d+1,l}) = d + 1$. Hence we have

$$\mathrm{VCdim}(\mathcal{F}_{d,a}) < d + 2.$$

Hence

$$\mathrm{VCdim}(\mathcal{F}_{d,a}) = d + 1.$$

□

3.4 Bounding the shattering coefficients from the VC-dimension

From the next lemma, we can bound the shattering coefficients from bounds on the VC-dimension.

Lemma 3.9 (Sauer lemma). *Let $\mathcal{F}$ be a non-empty set of functions from $[0,1]^d$ to $\{0,1\}$. Assume that* $\mathrm{VCdim}(\mathcal{F}) < \infty$. *Then we have, for $n \in \mathbb{N}$,*

$$\Pi_{\mathcal{F}}(n) \leq \sum_{i=0}^{\mathrm{VCdim}(\mathcal{F})} \binom{n}{i} \leq (n+1)^{\mathrm{VCdim}(\mathcal{F})},$$

with

$$\binom{n}{i} = \begin{cases} \frac{n!}{i!(n-i)!} \text{ if } i \in \{0,\dots,n\} \\ 0 \text{ if } i > n \end{cases}.$$

Proof of Lemma 3.9

For any set A, with H a non-empty set of functions from A to $\{0,1\}$, we can define $\Pi_H(n)$ and $\mathrm{VCdim}(H)$ in the same way as when $A = [0,1]^d$. Let us show

$$\text{For any set } A, \text{ for any set } H \text{ of functions from } A \text{ to } \mathbb{R}: \tag{3.2}$$
$$\Pi_H(k) \leq \sum_{i=0}^{V_H} \binom{k}{i}, \text{for } k = 1,\dots,n \text{ with } V_H = \mathrm{VCdim}(H).$$

We will show (3.2) by induction on k.

Let us show it for $k = 1$. If $V_H = 0$ then $\Pi_H(1) < 2^1 = 2$. Hence

$$\Pi_H(1) \leq 1 = \binom{1}{0} = \sum_{i=0}^{0} \binom{k}{i}.$$

Hence (3.2) is proved for $k = 1$ and $V_H = 0$.

If $V_H \geq 1$ we have

$$\Pi_H(1) = 2^1 = 2 = \binom{1}{0} + \binom{1}{1} \leq \sum_{i=0}^{V_H} \binom{k}{i}.$$

Hence eventually (3.2) is true for $k = 1$. Assume now that (3.2) is true for any k from 1 to $n-1$.

If $V_H = 0$ then there does not exist any $x \in A$ and $h_1, h_2 \in H$ such that $h_1(x) = 0$ and $h_2(x) = 1$ because for all $x \in A$, $\mathrm{card}\{h(x); h \in H\} < 2^1$. Hence for all $x_1,\dots,x_n \in A$,

$$\mathrm{card}\,\{(h(x_1),\dots,h(x_n)); h \in H\} = 1.$$

Hence

$$\Pi_H(n) = 1 = \sum_{i=0}^{0} \binom{n}{i}.$$

It thus remains to address the case $V_H \geq 1$.

For $x_1,\dots,x_n \in A$, define

$$H(x_1,\dots,x_n) = \{(h(x_1),\dots,h(x_n)); h \in H\}.$$

There exist $x_1, \dots, x_n \in A$ such that

$$\mathrm{card}(H(x_1, \dots, x_n)) = \Pi_H(n).$$

The set $H(x_1, \dots, x_n)$ only depend on the values of the functions in H on $\{x_1, \dots, x_n\}$. Hence, replacing

- A by $A = \{x_1, \dots, x_n\}$,
- H by

$$\tilde{H} = \{h' : \{x_1, \dots, x_n\} \to \{0,1\}; \text{there exists } h \in H \text{ such that } h'(x_i) = h(x_i) \text{ for } i = 1, \dots, n\},$$

we have $\Pi_H(n) = \Pi_{\tilde{H}}(n)$.

Hence in the sequel we assume that $A = \{x_1, \dots, x_n\}$ and H is a set of functions from $\{x_1, \dots, x_n\}$ to $\{0,1\}$, without loss of generality.

Let us consider the set

$$H' = \left\{h \in H; h(x_n) = 1 \text{ and } h' = h - \mathbf{1}_{\{x_n\}} \in H\right\},$$

composed of the functions that are equal to 1 at x_n and that stay in H if their value at x_n is replaced by 0. Notice that we have written $\mathbf{1}_{\{x_n\}} : \{x_1, \dots, x_n\} \to \{0,1\}$ defined by $\mathbf{1}_{\{x_n\}}(x_i) = \mathbf{1}_{x_n = x_i}$ for $i = 1, \dots, n$.

We use the notation, for a set G of functions from $\{x_1, \dots, x_n\}$ to $\{0,1\}$, and $\{x_{i_1}, \dots, x_{i_q}\} \subset \{x_1, \dots, x_n\}$,

$$G(x_{i_1}, \dots, x_{i_q}) = \{(g(x_{i_1}), \dots, g(x_{i_q})); g \in G\}.$$

We have

$$H(x_1, \dots, x_n) = H'(x_1, \dots, x_n) \cup (H \backslash H')(x_1, \dots, x_n)$$

and thus

$$\mathrm{card}H(x_1, \dots, x_n) \leq \mathrm{card}H'(x_1, \dots, x_n) + \mathrm{card}(H \backslash H')(x_1, \dots, x_n).$$

Step 1: bounding $\mathrm{card}H'(x_1, \dots, x_n)$

We observe that

$$\mathrm{card}H'(x_1, \dots, x_n) = \mathrm{card}H'(x_1, \dots, x_{n-1})$$

because $h(x_n) = 1$ for $h \in H'$.

If $q \in \mathbb{N}$ is such that there exists $\{x_{i_1}, \dots, x_{i_q}\} \subset \{x_1, \dots, x_n\}$ with $\mathrm{card}H'(x_{i_1}, \dots, x_{i_q}) = 2^q$ then $x_n \notin \{x_{i_1}, \dots, x_{i_q}\}$ (because $h(x_n) = 1$ for $h \in H'$). Also, we have $\mathrm{card}H(x_{i_1}, \dots, x_{i_q}, x_n) = 2^{q+1}$ because

$$2^{q+1} = \mathrm{card}\left(\{0,1\}^q \times \{0,1\}\right) = \mathrm{card}\left\{(h(x_{i_1}), \dots, h(x_{i_q}), h(x_n)); h \in H\right\}$$

by definition of H'. Hence $V_H \geq q + 1$ and thus $V_H \geq V_{H'} + 1$ (since q can be taken as $V_{H'}$). Hence $V_{H'} \leq V_H - 1$. Hence, we have, applying (3.2) with $k = n - 1$,

$$\mathrm{card}H'(x_1, \dots, x_n) = \mathrm{card}H'(x_1, \dots, x_{n-1}) \leq \Pi_{H'}(n-1) \leq \sum_{i=0}^{V_{H'}} \binom{n-1}{i} \leq \sum_{i=0}^{V_H - 1} \binom{n-1}{i}.$$

Step 2: bounding $\mathrm{card}(H \backslash H')(x_1, \dots, x_n)$

If $h, h' \in H\backslash H'$ satisfy $h(x_i) = h'(x_i)$ for $i = 1, \ldots, n-1$, then we can not have $h(x_n) \neq h'(x_n)$ (otherwise h or h' takes value 1 at x_n and thus belongs to H'). Hence we have

$$\mathrm{card}(H\backslash H')(x_1, \ldots, x_n) = \mathrm{card}(H\backslash H')(x_1, \ldots, x_{n-1}).$$

Also $V_{H\backslash H'} \leq V_H$ because $H\backslash H' \subset H$. Hence, using (3.2) with $k = n-1$ we have

$$\mathrm{card}(H\backslash H')(x_1, \ldots, x_{n-1}) \leq \Pi_{H\backslash H'}(n-1) \leq \sum_{i=0}^{V_{H\backslash H'}} \binom{n-1}{i} \leq \sum_{i=0}^{V_H} \binom{n-1}{i}.$$

Combining the two steps, we obtain

$$\begin{aligned}
\mathrm{card}H(x_1, \ldots, x_n) &\leq \sum_{i=0}^{V_H - 1} \binom{n-1}{i} + \sum_{i=0}^{V_H} \binom{n-1}{i} \\
&\leq \sum_{i=1}^{V_H} \binom{n-1}{i-1} + \sum_{i=0}^{V_H} \binom{n-1}{i} \\
&= 1 + \sum_{i=1}^{V_H} \left(\binom{n-1}{i-1} + \binom{n-1}{i} \right) \\
&= 1 + \sum_{i=1}^{V_H} \binom{n}{i} \\
&= \sum_{i=0}^{V_H} \binom{n}{i}.
\end{aligned}$$

We recall that $\mathrm{card}H(x_1, \ldots, x_n) = \Pi_H(n)$ and that we had started with any $A \subset [0,1]^d$ and any set of functions from A to $\{0,1\}$. Hence (3.2) is shown by induction. Hence we have

$$\Pi_{\mathcal{F}}(n) \leq \sum_{i=0}^{\mathrm{VCdim}(\mathcal{F})} \binom{n}{i}$$

which gives the first inequality of the lemma. For the second inequality, we have

$$\begin{aligned}
\sum_{i=0}^{\mathrm{VCdim}(\mathcal{F})} \binom{n}{i} &= \sum_{i=0}^{\min(\mathrm{VCdim}(\mathcal{F}), n)} \binom{n}{i} \\
&\leq \sum_{i=0}^{\min(\mathrm{VCdim}(\mathcal{F}), n)} \frac{n^i}{i!} \\
&\leq \sum_{i=0}^{\min(\mathrm{VCdim}(\mathcal{F}), n)} n^i \binom{\mathrm{VCdim}(\mathcal{F})}{i} \\
&\leq \sum_{i=0}^{\mathrm{VCdim}(\mathcal{F})} n^i \binom{\mathrm{VCdim}(\mathcal{F})}{i} \\
&= (n+1)^{\mathrm{VCdim}(\mathcal{F})},
\end{aligned}$$

using the Newton binomial formula at the end, which shows the second inequality of the lemma. □

Finally using Proposition 3.4 and Lemma 3.9, we obtain, for a set of functions $\mathcal{F}$ from $[0,1]^d$ to $\{0,1\}$,

$$\begin{aligned}\mathbb{E}\left(\sup_{f\in\mathcal{F}}\left|\mathbb{P}\left(f(X)\neq Y\right)-\frac{1}{n}\sum_{i=1}^{n}\mathbf{1}_{f(X_i)\neq Y_i}\right|\right) &\leq 2\sqrt{\frac{2\log(2\Pi_{\mathcal{F}}(n))}{n}}\\ &\leq 2\sqrt{\frac{2\log(2(n+1)^{\mathrm{VCdim}(\mathcal{F})})}{n}}\\ &= 2\sqrt{\frac{2\log(2)+2\mathrm{VCdim}(\mathcal{F})\log(n+1)}{n}}\end{aligned}$$

When $\mathrm{VCdim}(\mathcal{F})<\infty$ the bound goes to zero at rate almost $1/\sqrt{n}$. If we use a set of functions $\mathcal{F}_n$ that depends on n (more complex if there are more observations), then the rate of convergence is almost $\sqrt{\mathrm{VCdim}(\mathcal{F}_n)}/\sqrt{n}$.

CHAPTER 4 VC-DIMENSION OF NEURAL NETWORKS

The purpose of this first part is to properly introduce the notations and the notions of stochastic programming and how randomness can intervene in optimization methods.

The section is based on (Bartlett et al., 2019).

4.1 Neural networks as directed acyclic graphs

We will use graphs. A *directed graph* is of the form (V, E), where V stands for *vertices* and E stands for *edges*. The set V is a finite set, for instance $V = \{v_1, \dots, v_n\}$ or $V = \{1, \dots, n\}$. The set E is a subset of $V \times V$ that does not contain any element of the form (v, v), $v \in V$.

If $(v_1, v_2) \in E$, we say that there is a *path* from v_1 to v_2. We say that v_1 is a *predecessor* of v_2 and that v_2 is a *successor* of v_1. A simple example is given in Figure 4.1.

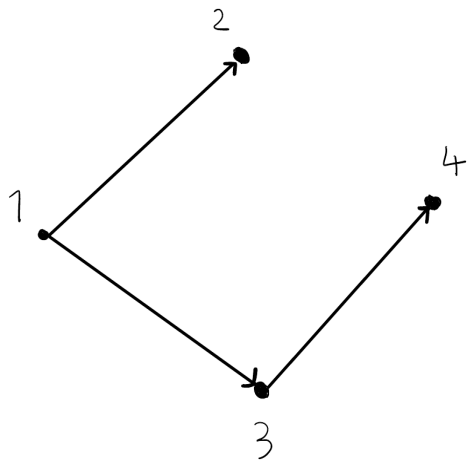

Figure 4.1: A directed graph defined by $V = \{1, 2, 3, 4\}$ and $E = \{(1, 2), (1, 3), (3, 4)\}$ with 4 vertices and 3 edges.

We say that the directed graph (V, E) is *acyclic* if there does not exist any $n \in \mathbb{N}$

and $v_1, \ldots, v_n \in V$ such that

- $v_n = v_1$,
- $(v_1, v_2), \ldots, (v_{n-1}, v_n) \in E$.

A simple example is given in Figure 4.2.

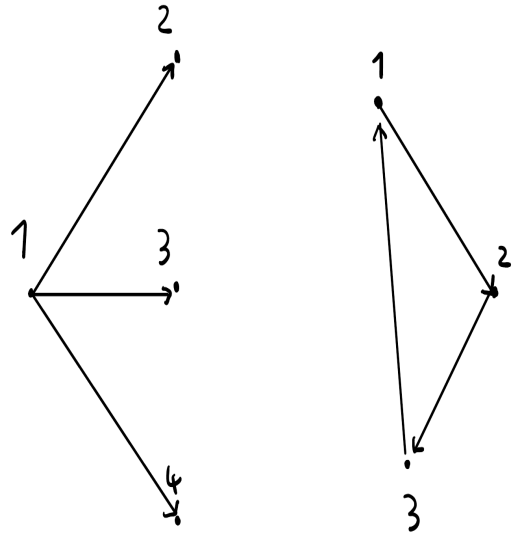

Figure 4.2: The graph on the left is acyclic and the graph on the right is cyclic (not acyclic).

A directed graph which is acyclic is called a *DAG* (directed acyclic graph). We call *path* a vector $(v_1, \ldots, v_n)$ with $v_1, \ldots, v_n \in V$ and $(v_1, v_2), \ldots, (v_{n-1}, v_n) \in E$. For a DAG (V, E) and $v \in V$ we call *indegree* of v the quantity $\mathrm{card}\{(v', v), v' \in V, (v', v) \in E\}$. We call *outdegree* of v the quantity $\mathrm{card}\{(v, v'), v' \in V, (v, v') \in E\}$. A simple example is given in Figure 4.3.

Definition 4.1 (general feed-forward neural network)**.** A feed-forward neural network is defined by the following.

- An activation function $\sigma : \mathbb{R} \to \mathbb{R}$.
- A DAG $G = (V, E)$ such that G has $d \geq 1$ vertices with indegree 0 and 1 vertex of outdegree 0. We write the d vertices with indegree 0 as
$$s_1^{(0)}, \ldots, s_d^{(0)}.$$
- A vector of weights
$$(w_a; a \in V' \cup E)$$
where V' is the set of vertices with non-zero indegrees (there is a weight per vertex [except the d vertices with indegree 0] and a weight per edge).

We write L for the maximal length (number of edges) of a path of G. We have $L \leq \mathrm{card}(V) - 1$. We define the layers $0, 1, \ldots, L$ by induction as follows.

- The layer 0 is the set {vertices with indegree 0}.

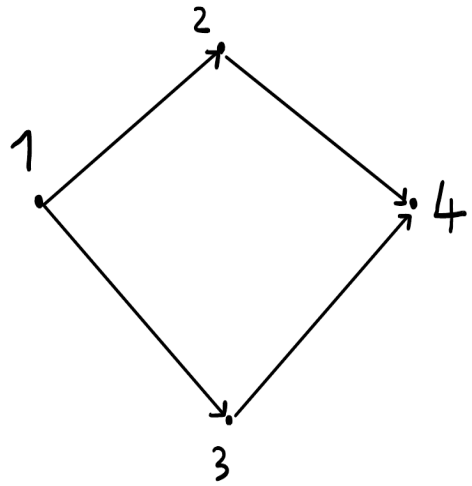

Figure 4.3: The vertex 1 has indegree 0 and outdegree 2. The vertex 4 has indegree 2 and outdegree 0.

- For $\ell = 1, \dots, L$,

$$\text{layer } \ell \ = \Big\{\text{vertices who have a predecessor in the layer } \ell - 1, \text{ possibly other predecessors in the layers } 0, 1, \dots, \ell - 2, \text{ and no other predecessors}\Big\}.$$

Proposition 4.2. *In the context of Definition 4.1, we have the following.*

- *The layers* $0, 1, \dots, L$ *are non-empty.*
- *The layers* $0, 1, \dots, L$ *are disjoint sets.*
- *Any vertex belongs to a layer.*
- *The layer* L *is a singleton composed of the unique vertex of outdegree* 0.
- *The edges are only of the form* (v, v'), *with* $v \in$ *layer* i *and* $v' \in$ *layer* j *with* $i < j$.

Proof of Proposition 4.2

We call the elements of the layer 0 the roots. For a vertex v, we call *inverse path* from v to the roots a vector $(v, v_1, \dots, v_k)$ with v_k a root and $(v_1, v), (v_2, v_1), \dots, (v_k, v_{k-1}) \in E$ (hence $(v_k, v_{k-1}, \dots, v_2, v)$ is a path). The length of such an inverse path is k (there are k edges in the path). By convention, if v is a root, we say that v has an inverse path of length 0 to the roots.

Then let us show by induction that, for $\ell = 0, \dots, L$,

$$\text{layer } \ell = \{ \text{ vertices which longest inverse paths to the roots have length } \ell\}\,. \tag{4.1}$$

The property is true for the layer 0 (with our convention).

If the property is true for the layer ℓ, then any vertex of the layer $\ell + 1$ has an edge that comes from the layer ℓ, so it has an inverse path to the roots of length $\ell + 1$. There are no longer inverse path because the vertex has no predecessors outside of the layers $0, 1 \dots, \ell$.

Consider a vertex v which longest inverse path to the roots has length $\ell+1$. The first predecessor of v in this path belongs to the layer ℓ, from (4.1) at step ℓ. The only predecessors of v are in the layers 0 to ℓ because if there are other predecessors, the longest inverse path from v to the roots has length $W > \ell$ (because this other predecessor would not belong to the layers 0 to ℓ and using (4.1)). Hence, finally, v belongs to the layer $\ell+1$. Hence we have shown (4.1) by induction.

We remark that any vertex v has an inverse path to the roots. Indeed, we let $\bar{V}$ be the subset of S composed of the vertices which have an inverse path to the roots. Then, is $\bar{V} \neq V$, then $V \backslash \bar{V}$ provides a DAG with one vertex of indegree 0 which is false (in a DAG, there exists a vertex of indegree zero, otherwise we can construct an arbitrary long inverse path and thus a cycle).

Hence, from (4.1), an element of the layer L has outdegree 0 (otherwise there is a path of length $L+1$). Hence, the layer L is empty or is a singleton. By construction of the layers, the edges only go from a layer i to a layer j with $i < j$. Hence, the only possible path of length L go through each of the layers $0, 1, \ldots, L$. Since such a path exists, the layers $0, 1, \ldots, L$ are non-empty.

Hence we have proved everything: the layers are non-empty, disjoint, any vertex belongs to one of the layers and the edges go from a layer to a layer of strictly larger index. □

An example is given in Figure 4.4.

Remark Compared to Section 1.3,

- we do not have all the possible edges between the layers i and $i+1$,
- we allow for edges between the layers i and $i+k$ with $k \geq 2$.

Formal definition of a general feed-forward neural network function based on a DAG

Following definition 4.1 and Proposition 4.2, it is a function characterized by

$$(w_a; a \in V' \cup E)$$

where V' contains the layers 1 to L. The input space is $[0,1]^d$. Consider an input $x = (x_1, \ldots, x_d) \in [0,1]^d$. We define by induction on the layers 0 to L the outputs associated to each neurons of the layer ℓ. For the layer 0 the output of the vertex $s_i^{(0)}$ is x_i.

For the layer $\ell+1$, $\ell = 0, \ldots, L-2$, the output of a vertex v is

$$\sigma\left(\sum_{i=1}^{m} w_i S_i + b\right)$$

where

- m is the indegree of v,
- $v'_1, \ldots, v'_m$ are the predecessors of v: $(v'_1, v), \ldots, (v'_m, v) \in E$,
- $(w_1, \ldots, w_m) = (w_{(v'_1,v)}, \ldots, w_{(v'_m,v)})$, the weights associated to the edges pointing to v,
- $S_1, \ldots, S_m$ are the outputs of $v_1, \ldots, v'_m$, which are vertices of the layers $0, \ldots, \ell$ so these outputs are indeed already defined,
- $b = w_v$ is the weight associated to the vertex v.

For the layer L, with the same notations, the output is

$$1_{\sum_{i=1}^{m} w_i S_i + b \geq 0}.$$

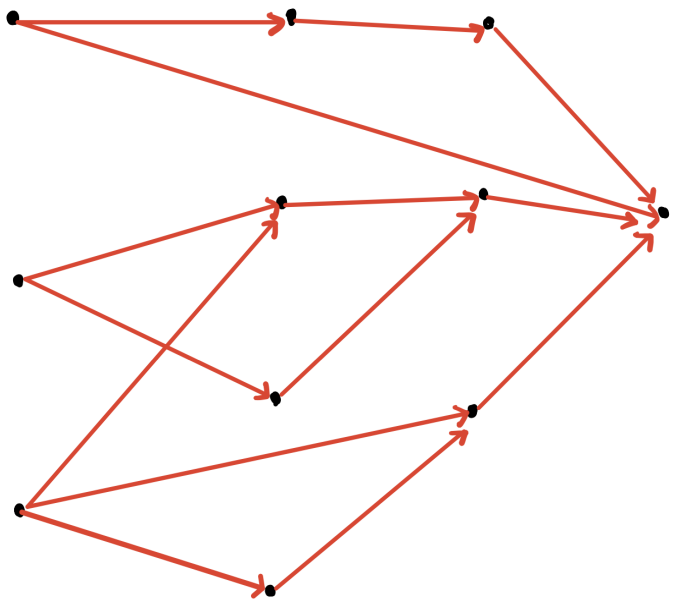

Figure 4.4: An example of the DAG of a neural network. The layer 0 has 3 vertices, representing a neural network classifier from $[0,1]^3$ to $\{0,1\}$. These vertices have indegree 0. The layer 1 has 4 vertices (neurons). The layer 2 has 3 vertices (neurons). The layer $L = 3$ has one (final) vertex of outdegree 0 (output of the neural network function). The layers 1 and 2 correspond to the hidden layers.

4.2 Bounding the VC-dimension

We will bound the VC-dimension of these neural network functions based on the following quantities.

- L: number of layers minus 1 (longest path).
- U: number of neurons. $U = \text{card}(V')$ where V' is the set of vertices of the layers 1 to L.
- W: number of weights. $W = U + \text{card}(E)$.

We assume that σ is piecewise polynomial: there exist $I_1, \ldots, I_{p+1}$ pieces ($p \geq 1$) where $I_1, \ldots, I_{p+1}$ are intervals of $\mathbb{R}$, that is of the form

$$(-\infty, a), (-\infty, a], (a, b), [a, b), (a, b], [a, b], (a, +\infty), [a, +\infty)$$

such that $I_i \cap I_j = \emptyset$ for $i \neq j$, with $\mathbb{R} = \cup_{i=1}^{p+1} I_i$ and such that σ is *polynomial* on I_i for $i = 1, \ldots, p+1$ with a polynomial function of degree smaller or equal to $D \in \mathbb{N}$.

Examples

- Threshold function $\sigma(x) = 1_{x \geq 0}$ with $p = 1$, $I_1 = (-\infty, 0)$, $I_2 = [0, +\infty)$ and $D = 0$. The polynomials are $x \mapsto 0$ on I_1 and $x \mapsto 1$ on I_2.

- ReLU function $\sigma(x) = \max(0, x)$ with $p = 1$, $I_1 = (-\infty, 0)$, $I_2 = [0, +\infty)$ and $D = 1$. The polynomials are $x \mapsto 0$ on I_1 and $x \mapsto x$ on I_2.

Theorem 4.3 ((Bartlett et al., 2019)). *Let $L \geq 1$, $U \geq 3$, $d \geq 1$, $p \geq 1$ and $W \geq U \geq L$. We consider a DAG $G = (V, E)$ which longest path has length L, with d vertices with indegree 0 and one vertex with outdegree 0. We assume that $\mathrm{card}(V') = U$ where V' is the set of vertices in the layers 1 to L. We assume that $U + \mathrm{card}(E) = W$.*
We consider a function $\sigma : \mathbb{R} \to \mathbb{R}$, piecewise polynomial on $p+1$ disjoint intervals, with degrees smaller or equal to D.
We define the following, for $i \in \{1, \dots, L\}$.

- *If $D = 0$, W_i is the number of parameters (weights and biases) useful to the computation of all the neurons of the layer i. We have*

  $W_i =$ *number of edges pointing to the layer i + number of vertices in the layer i.*

- *If $D \geq 1$, W_i is the number of parameters (weights and biases) useful to the computation of all the neurons of the layers 1 to i. We have*

  $W_i =$ *number of edges pointing to a layer j, $j \leq i$ + number of vertices in the layers 1 to i .*

We write

$$\bar{L} = \frac{1}{W} \sum_{i=1}^{L} W_i \quad \in [1, L],$$

- *this is equal to 1 if $D = 0$,*
- *this can be close to L if $D \geq 1$ and if the neurons are concentrated on the first layers.*

We define, for $i = 1, \dots, L$, k_i as the number of vertices of the layer i ($k_L = 1$). We write

$$R = \underbrace{\sum_{i=1}^{L} k_i (1 + (i-1) D^{i-1})}_{\leq U L D^{L-1}} \quad \text{if } D \geq 1$$

and

$$R = U \quad \text{if } D = 0.$$

We define $\mathcal{F}$ as the set of all the feed-forward neural networks defined by $G = (V, E)$, with one weight per vertex of the layers 1 to L and one weight per edge (the structure of the network is fixed and the weights are varying).
Then, for $m \geq W$, with $e = \exp(1)$,

$$\Pi_{\mathcal{F}}(m) \leq \prod_{i=1}^{L} 2 \left(\frac{2emk_i p(1 + (i-1)D^{i-1})}{W_i} \right)^{W_i} \tag{4.2}$$

$$\leq \left(4emp(1 + (L-1)D^{L-1}) \right)^{\sum_{i=1}^{L} W_i}. \tag{4.3}$$

Furthermore

$$\mathrm{VCdim}(\mathcal{F}) \leq L + \bar{L} W \log_2 \left(4epR \log_2 (2epR) \right). \tag{4.4}$$

In particular we have the following.

- If $D = 0$, $\mathrm{VCdim}(\mathcal{F}) \leq L + W \log_2\left(4epU \log_2(2epU)\right)$ has W as a dominating term (neglecting logarithms). This is the number of parameters of the neural network functions.
- If $D \geq 1$, $\mathrm{VCdim}(\mathcal{F})$ has $\bar{L}W$ as a dominating term (neglecting logarithms). This is more than the number of parameters of the neural network functions. We can interpret this by the fact that depth can increase $\bar{L}$ (recall that $\bar{L} \in [1, L]$) and thus make the family of neural network functions more complex.

4.3 Proof of the theorem

Let us prove Theorem 4.3. The proof relies of the following result from algebraic geometry.

Lemma 4.4. *Let $P_1, \ldots, P_m$ be polynomials functions of $n \leq m$ variables of degree smaller or equal to $D \geq 1$. We write*

$$K = \mathrm{card}\left\{(\mathrm{sign}(P_1(x)), \ldots, \mathrm{sign}(P_m(x))) \, ; \, x \in \mathbb{R}^n\right\},$$

with $\mathrm{sign}(t) = \mathbf{1}_{t \geq 0}$. Note that K is the number of possible sign vectors. Then

$$K \leq \left(\frac{2emD}{n}\right)^n.$$

The proof of Lemma 4.4 can be found in (Anthony and Bartlett, 2009).

Let us write $f(x, a)$ for the output of the network (without the indicator function at the end) for the input $x \in [0,1]^d$ and the vector of parameters $a \in \mathbb{R}^W$. Let $x_1, \ldots, x_m \in [0,1]^d$. In order to bound $\Pi_{\mathcal{F}}(m)$, let us bound

$$\begin{aligned}
&\mathrm{card}\left\{(\mathrm{sign}(f(x_1, a)), \ldots, \mathrm{sign}(f(x_m, a))) \, ; \, a \in \mathbb{R}^W\right\} \\
&\leq \sum_{i=1}^{N} \mathrm{card}\left\{(\mathrm{sign}(f(x_1, a)), \ldots, \mathrm{sign}(f(x_m, a))) \, ; \, a \in P_i\right\},
\end{aligned}$$

where $P_1, \ldots, P_N$ are a partition of $\mathbb{R}^W$ which will be chosen such that the m functions $a \mapsto f(x_j, a)$, $j = 1, \ldots, m$, are polynomial on each cell P_i. We can then apply Lemma 4.4.

The main difficulty is to construct a good partition. We will construct by induction partitions $C_0, \ldots, C_{L-1}$, where C_{L-1} will be the final partition $P_1, \ldots, P_N$.

The partitions $C_0, \ldots, C_{L-1}$ will be partitions of $\mathbb{R}^W$ such that for $i \in \{0, \ldots, L-1\}$, $C_i = \{A_1, \ldots, A_q\}$ with $A_1 \cup \cdots \cup A_q = \mathbb{R}^W$ and $A_r \cup A_{r'} = \emptyset$ for $r \neq r'$. We will have the following.

(a) The partitions are nested, any $C \in C_i$ is a union of one or several $C' \in C_{i+1}$ $(0 \leq i \leq L-2)$.

(b) We have $\mathrm{card}(C_0) = 1$ ($C_0 = \{\mathbb{R}^W\}$) and for $i \in \{1, \ldots, L-1\}$,

$$\frac{\mathrm{card}(C_i)}{\mathrm{card}(C_{i-1})} \leq 2\left(\frac{2emk_ip(1 + (i-1)D^{i-1})}{W_i}\right)^{W_i}.$$

(c) For $i \in \{0, \ldots, L-1\}$, for $E \in C_i$, for $j \in \{1, \ldots, m\}$, the output of a neuron of the layer i (for the input x_j) is a polynomial function of W_i variables of $a \in E$, with degree smaller or equal to iD^i.

Induction

When $i = 0$, we have $C_0 = \{\mathbb{R}^W\}$. The output of a neuron of the layer 0 is constant with respect to $a \in \mathbb{R}^W$ and thus the property (c) holds.

Let $1 \leq i \leq L-1$. Assume that we have constructed nested partitions $C_0, \ldots, C_{i-1}$ satisfying (b) and(c). Let us construct C_i.

We write $P_{h,x_j,E}(a)$ the input (just before σ) of the neuron h ($h = 1, \ldots, k_i$) of the layer i, for the input x_j, as a function of $a \in E$ with $E \in C_{i-1}$. From the induction hypothesis (c), since $P_{h,x_j,E}(a)$ is of the form

$$\sum_k w_k (\text{output of neuron } k) + b$$

and since the partitions are nested, we have that $P_{h,x_j,E}(a)$ is polynomial on E of degree smaller of equal to $1 + (i-1)D^{i-1}$ and depends at most on W_i variables (we can check that this holds also when $D = 0$).

Because of σ, the output of the neuron h is piecewise polynomial on E. We will divide E into subcells such that the output is polynomial on each of the subcells, for any neurons h and any input x_j. Figure 4.5 illustrates the current state of the proof.

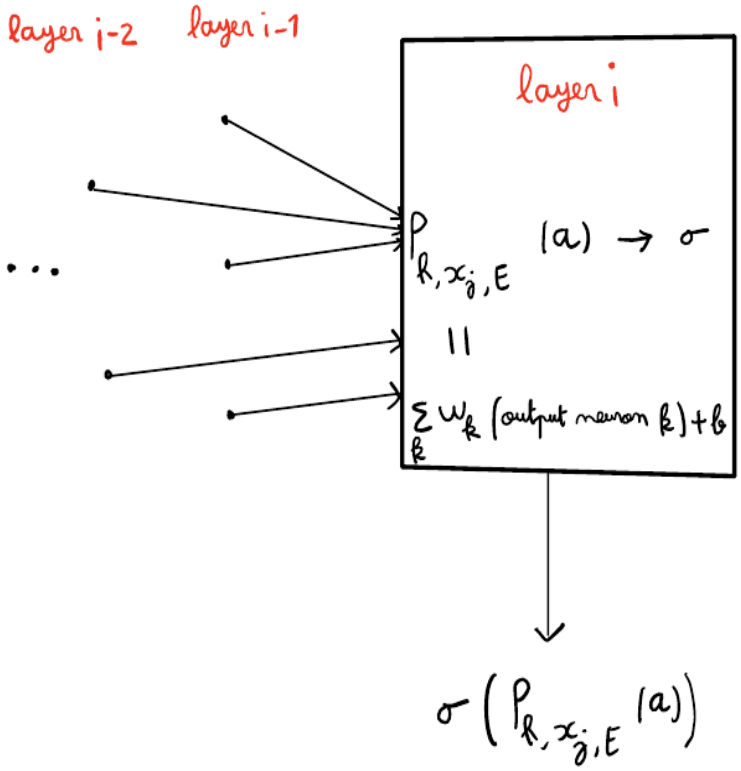

Figure 4.5: Illustration of the construction of the partitions.

We write $t_1 < t_2 < \cdots < t_p$ the cuts of the pieces $I_1, \ldots, I_{p+1}$, as illustrated in Figure 4.6.

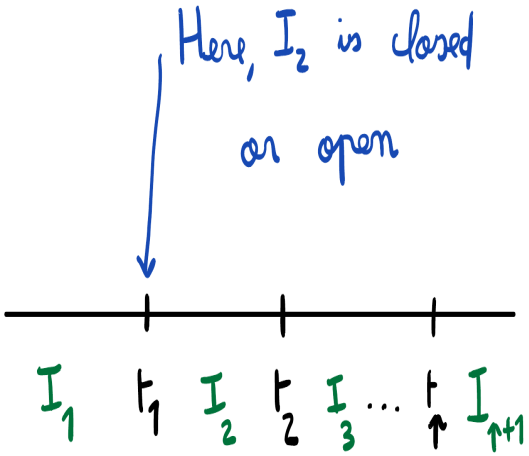

Figure 4.6: Illustration of the cuts of the intervals for σ.

We consider the polynomials

$$\pm \left(P_{h,x_j,E}(a) - t_r \right)_{\substack{h \in \{1,\ldots,k_i\} \\ j \in \{1,\ldots,m\} \\ r \in \{1,\ldots,p\}}},$$

where in the above display there is a $+$ if I_{r+1} is closed at t_r and a $-$ if I_{r+1} is open at t_r. With this,

$$\mathbf{1}_{\pm\left(P_{h,x_j,E}(a)-t_r\right)\geq 0} = \operatorname{sign}\left(\pm \left(P_{h,x_j,E}(a) - t_r \right) \right)$$

is constant for $P_{h,x_j,E}(a) \in I_{r+1}$.

From Lemma 4.4, this set of polynomials on $\mathbb{R}^W$ reaches at most

$$\Pi = 2 \left(\frac{2e(k_i mp)(1 + (i-1)D^{i-1})}{W_i} \right)^{W_i}$$

distinct vectors of signs, $\operatorname{sign}\left(\pm \left(P_{h,x_k,E}(a) - t_r \right) \right)_{h,j,r}$ when $a \in \mathbb{R}^W$ and thus when $a \in E$. Indeed,

- $k_i mp$ is the number of polynomials,
- $1 + (i-1)D^{i-1}$ is the degree bound,
- W_i is the number of variables.

We can thus partition E into less than Π subcells such that, on each of these subcells, the $P_{h,x_j,E}(a)$ stay in the same interval where σ is polynomial as a varies in the subcell. We remark that these Π subcells of E are the same for all the neurons h and all the inputs x_j (this is important for the sequel).

Hence we obtain a new partition C_i of cardinality less than $\Pi \text{card}(C_{i-1})$. This enables to satisfy the property (b).

Let us now address the property (c). For all $E' \in C_i$, the output of the neuron $h \in \{1, \dots, k_i\}$,

$$a \in E' \mapsto \sigma\left(P_{h,x_j,E}(a)\right)$$

is a polynomial function of W_i variables with degree smaller or equal to

$$D(1 + (i-1)D^{i-1}) \leq iD^i,$$

where the factor D comes from the application of the polynomial corresponding to σ. Hence the property (c) holds.

This completes the induction and we have the nested partitions $C_0, \dots, C_{L-1}$ satisfying (b) and (c).

Use of the partition to conclude the proof

In particular, C_{L-1} is a partition of $\mathbb{R}^W$ such that the output of each neuron of the layers $0, \dots, L-1$ is polynomial of degree smaller or equal to $(L-1)D^{L-1}$ on each $E \in C_{L-1}$ (since the partitions are nested) and for all input $x_1, \dots, x_m$.

Hence for each cell $E \subset C_{L-1}$ and each input x_j, the function

$$a \in E \mapsto f(x_j, a)$$

at the end of the network is polynomial with degree less of equal to $1 + (L-1)D^{L-1}$ where the 1 comes from the final linear combination.

Hence, from Lemma 4.4,

$$\text{card}\left\{(\text{sign}(f(x_1, a)), \dots, \text{sign}(f(x_m, a))) \, ; \, a \in E\right\} \leq 2\left(\frac{2em(1 + (L-1)D^{L-1})}{W_L}\right)^{W_L}$$

and thus

$$\begin{aligned}\text{card}\left\{(\text{sign}(f(x_1, a)), \dots, \text{sign}(f(x_m, a))) \, ; \, a \in \mathbb{R}^w\right\} &\leq \sum_{E \in C_{L-1}} \text{card}\left\{(\text{sign}(f(x_1, a)), \dots, \text{sign}(f(x_m, a))) \, ; \, a \in E\right\} \\ &\leq \text{card}(C_{L-1}) 2\left(\frac{2em(1 + (L-1)D^{L-1})}{W_L}\right)^{W_L}.\end{aligned}$$

Then, from the property (b),

$$\text{card}(C_{L-1}) \leq \prod_{i=1}^{L-1} 2\left(\frac{2emk_ip(1 + (i-1)D^{i-1})}{W_i}\right)^{W_i}$$

and thus, since (**??**) holds for any $x_1, \dots, x_m \in [0,1]^d$,

$$\Pi_{\mathcal{F}}(m) \leq \prod_{i=1}^{L} 2\left(\frac{2emk_ip(1 + (i-1)D^{i-1})}{W_i}\right)^{W_i}$$

and thus (4.2) is proved.

For the sequel, we use the inequality between arithmetic and geometric means: for $y_1, \dots, y_k > 0$, for $a_1, \dots, a_k \geq 0$ such that $\sum_{i=1}^k a_i > 0$,

$$\prod_{i=1}^{k} y_i^{a_i} \leq \left(\frac{\sum_{i=1}^k a_i y_i}{\sum_{i=1}^k a_i}\right)^{\sum_{i=1}^k a_i}.$$

Then we have

$$\begin{aligned}
\Pi_{\mathcal{F}}(m) &\leq 2^L \left(\frac{2emp \sum_{i=1}^{L} k_i (1 + (i-1)D^{i-1})}{\sum_{i=1}^{L} W_i} \right)^{\sum_{i=1}^{L} W_i} \\
\text{(by definition of } R\text{:)} \quad &= 2^L \left(\frac{2empR}{\sum_{i=1}^{L} W_i} \right)^{\sum_{i=1}^{L} W_i} \\
\text{(since } L \leq \sum_{i=1}^{L} W_i\text{:)} \quad &\leq \left(\frac{4emp(1 + (L-1)D^{L-1}) \sum_{i=1}^{L} k_i}{\sum_{i=1}^{L} W_i} \right)^{\sum_{i=1}^{L} W_i} \\
\text{(since } \sum_{i=1}^{L} k_i \leq \sum_{i=1}^{L} W_i\text{:)} \quad &\leq \left(4emp(1 + (L-1)D^{L-1}) \right)^{\sum_{i=1}^{L} W_i}.
\end{aligned}$$

Hence (4.3) is proved.

To prove the bound (4.4) on $\mathrm{VCdim}(\mathcal{F})$ we will combine (**??**) and the next lemma (that we do not prove).

Lemma 4.5. *Let $r \geq 16$ and $w \geq t > 0$. Then, for any $m > t + w \log_2(2r \log_2(r)) := x_0$, we have*

$$2^m > 2^t \left(\frac{mr}{w} \right)^w.$$

Hence from (**??**) and by definition of the VC-dimension, Lemma 4.5 with $t = L$, $w = \sum_{i=1}^{L} W_i$ and

$$r = 2epR \geq 2eU \geq 16$$

yields

$$\mathrm{VCdim}(\mathcal{F}) \leq L + \left(\sum_{i=1}^{L} W_i \right) \log_2 \left(4epR \log_2(2epR) \right)$$

which proves (4.4).

CHAPTER 5 GENERALIZATION ERROR FOR REGRESSION WITH THE METRIC ENTROPY

THE purpose of this chapter is to understand the role of the estimators' class in terms of generalization for classification problems.

We have seen that the *approximation error* can be (very) small in modern models, notably with deep networks. Now, we adopt a complementary vision by looking at the generalization error. Here, we focus on regression tasks. More precisely, we investigate the term

$$\mathbb{E}\left(\sup_{f\in\mathcal{F}}\left|\mathbb{E}\left((f(X)-Y)^2\right)-\frac{1}{n}\sum_{i=1}^{n}(f(X_i)-Y_i)^2\right|\right)$$

5.1 COVERING AND METRIC ENTROPY

For a function class $\mathcal{F}$ equipped with the L_∞ metric, the *covering number*

$$N(\varepsilon,\mathcal{F},L_\infty)$$

is the smallest number of L_∞-balls of radius ε needed to cover $\mathcal{F}$. The *metric entropy* is defined as

$$\mathcal{H}(\varepsilon,\mathcal{F},L_\infty)=\log N(\varepsilon,\mathcal{F},L_\infty).$$

It measures the complexity of $\mathcal{F}$ at scale ε.

Metric-entropy of one-dimensional linear predictors

Setup. We consider the class of one-dimensional linear predictors from $[0,1]\to\mathbb{R}$

$$\mathcal{F}_B^{(1)} \;=\; \{f_w:[0,1]\to\mathbb{R},\; f_w(x)=wx \;:\; |w|\le B\}$$

where $B>0$ is a bounded radius.

We measure distances with the uniform (sup) norm on X, $\|f\|_\infty=\sup_{x\in X}|f(x)|$.

Proposition 5.1. *For every $\varepsilon>0$,*

$$N\big(\varepsilon,\mathcal{F}_B^{(1)},\|\cdot\|_\infty\big) \;\le\; 1+\frac{2B}{\varepsilon},$$

and therefore

$$\log N(\varepsilon,\mathcal{F}_B^{(1)},\|\cdot\|_\infty) \;\le\; \log\Big(1+\frac{2B}{\varepsilon}\Big).$$

Proof. For two parameters $w, w' \in [-B, B]$ and any $x \in [0, 1]$,

$$|f_w(x) - f_{w'}(x)| = |(w - w')x| \le |w - w'|\,|x| \le |w - w'|.$$

Hence the map $w \mapsto f_w$ is 1-Lipschitz from the parameter interval $[-B, B]$ (with absolute value norm) to $(C(X), \|\cdot\|_\infty)$.
Let $\delta > 0$ be a mesh size to be chosen below. Cover the parameter interval $[-B, B]$ by one-dimensional intervals of length δ. The number M of such intervals is

$$M = \left\lceil \frac{2B}{\delta} \right\rceil \le 1 + \frac{2B}{\delta}.$$

If $|w - w'| \le \delta$ then $\|f_w - f_{w'}\|_\infty \le \delta$. Choosing $\delta = \varepsilon$ yields an ε-cover of $\mathcal{F}_B^{(1)}$ with at most $1 + 2B/\delta = 1 + 2B/\varepsilon$ elements. That proves the stated bound. □

Remarks. The bound is parametric: $\log N(\varepsilon) \asymp \log(1/\varepsilon)$ with constant proportional to BR. It is tight in order: one cannot generally improve the logarithmic dependence on $1/\varepsilon$ for a one-parameter family.

Metric-entropy of d-dimensional linear predictors

Setup. Let $X \subset \mathbb{R}^d$ be a bounded input domain and set

$$R := \sup_{x \in X} \|x\|_2 < \infty.$$

Consider the class of linear predictors with bounded Euclidean norm

$$\mathcal{F}_B^{(d)} = \{f_w(x) = w^\top x : w \in \mathbb{R}^d,\ \|w\|_2 \le B\}.$$

Distances are measured in the sup-norm on X, $\|f\|_\infty = \sup_{x \in X} |f(x)|$.

Proposition 5.2. *For every $\varepsilon > 0$,*

$$N\big(\varepsilon, \mathcal{F}_B^{(d)}, \|\cdot\|_\infty\big) \ \le\ \left(1 + \frac{2BR}{\varepsilon}\right)^d,$$

and hence

$$\log N(\varepsilon, \mathcal{F}_B^{(d)}, \|\cdot\|_\infty) \ \le\ d \log\left(1 + \frac{2BR}{\varepsilon}\right) \asymp d \log \frac{BR}{\varepsilon}.$$

Proof. For any two parameter vectors $w, w' \in \mathbb{R}^d$ and any $x \in X$, by Cauchy–Schwarz

$$|f_w(x) - f_{w'}(x)| = |(w - w')^\top x| \le \|w - w'\|_2\, \|x\|_2 \le R\, \|w - w'\|_2.$$

Thus the map $w \mapsto f_w$ is R-Lipschitz from $(\mathbb{R}^d, \|\cdot\|_2)$ to $(C(X), \|\cdot\|_\infty)$.
Consequently an ε-cover of $\mathcal{F}_B^{(d)}$ in $\|\cdot\|_\infty$ is obtained by any δ-cover of the parameter ball $B_2^d(B) = \{w : \|w\|_2 \le B\}$ in Euclidean norm with $\delta = \varepsilon/R$. Standard volumetric (grid) coverings of the Euclidean ball yield

$$N(\delta, B_2^d(B), \|\cdot\|_2) \le \left(1 + \frac{2B}{\delta}\right)^d.$$

Plugging $\delta = \varepsilon/R$ gives

$$N(\varepsilon, \mathcal{F}_B^{(d)}, \|\cdot\|_\infty) \le \left(1 + \frac{2B}{\varepsilon/R}\right)^d = \left(1 + \frac{2BR}{\varepsilon}\right)^d,$$

which is the claimed bound. □

Remarks.

- The factor d in the entropy (logarithm) reflects the intrinsic (parameter) dimension of the class: $\log N \asymp d \log(1/\varepsilon)$.

5.2 Consequences on the generalization

Theorem 5.3 (Uniform deviation bound, expectation form)**.** *There exists an absolute constant C such that for any class $\mathcal{F} \subseteq \{f : [0,1]^d \to [-1,1]\}$,*

$$\mathbb{E}\left(\sup_{f\in\mathcal{F}}\left|\mathbb{E}\left((f(X)-Y)^2\right) - \frac{1}{n}\sum_{i=1}^{n}(f(X_i)-Y_i)^2\right|\right) \leq C \inf_{\varepsilon>0}\left\{\varepsilon + \sqrt{\frac{\log N(\varepsilon,\mathcal{F},L_\infty)}{n}}\right\}.$$

5.3 Conclusion

This means that the generalization error is here controlled by the metric entropy. The higher it is, the more involved the function class, the harder the learning phase... but the lower the approximation error. Universal bounds difficult to obtain as the complexity of the Bayes predictor is unknown a priori.

For neural networks, we have

$$\log N(\varepsilon, NN, L_\infty) \approx W \log\left(\frac{CB}{\varepsilon}\right),$$

where W denotes the total number of scalar parameters of the network, B is a bound on parameter magnitude, and C is a universal constant.

CHAPTER 6 Optimization of neural networks

6.1 Backpropagation for neural networks

In Section 6.1, we assume that $\sigma : \mathbb{R} \to \mathbb{R}$ is differentiable with derivative function σ'.

Motivation.

Here we consider the neuron network output with several hidden layers. We fix the architecture, the activation function σ and the input $x \in [0,1]^d$. This output depends on the parameters θ gathering all the weights and biases:

$$\theta = \left(v, b_1^{(c)}, \dots, b_{N_c}^{(c)}, w_1^{(c)}, \dots, w_{N_c}^{(c)}, \dots, b_1^{(1)}, \dots, b_{N_1}^{(1)}, w_1^{(1)}, \dots, w_{N_1}^{(1)}\right).$$

We write $f_\theta(x)$ for the neural network output.

Our aim is to compute the gradient of $f_\theta(x)$ with respect to θ. This is very useful when the parameters θ of a neural network are optimized, for instance with least squares in regression with

$$\min_\theta \frac{1}{n} \sum_{i=1}^{n} (f_\theta(X_i) - Y_i)^2 .$$

To compute the gradient of this function with respect to θ, it is sufficient to compute the gradients of $f_\theta(X_i)$ with respect to θ for $i = 1, \dots, n$.

Backpropagation algorithm.

At first sight, computing the gradient of $f_\theta(x)$ with respect to θ is very challenging because of the compositions of functions in (1.1). Even in the simpler case where σ is the identity, if we expand (1.1) into a sum of products, the number of terms in the sum will be of order $N_1 \times \cdots \times N_c$ which can be of astronomical order for deep networks.

The backpropagation algorithm presented here solves this issue by enabling to compute the gradient with storages and operations of complexity at most $N_k N_{k+1}$ for $k \in \{1, \dots, c-1\}$.

We define the vector $\eta_\theta^{(c)}$ of dimension $1 \times N_c$ equal to

$$\eta_\theta^{(c)} = (\sigma'(g_{\theta,1}^{(c)}(x))v_1, \dots, \sigma'(g_{\theta,N_c}^{(c)}(x))v_{N_c})$$

where $g_\theta^{(c)}(x) = (g_{\theta,1}^{(c)}(x), \ldots, g_{\theta,N_c}^{(c)}(x))$ and

$g_\theta^{(c)}(x)$ is the vector of size N_c composed by the values at layer c **just before** the activations σ.

Lemma 6.1. *$\eta_\theta^{(c)}$ is the (line) gradient of the network output with respect to the vector of values at layer c **just before** the activations σ.*

In Lemma 6.1, more precisely, the output is a function of $g_\theta^{(c)}(x)$ and v, and we take the gradient with respect to $g_\theta^{(c)}(x)$ at $g_\theta^{(c)}(x)$ for v fixed.

Proof of Lemma 6.1 The output of the network is a function of $g_\theta^{(c)}(x)$ and the final weights $v_1, \ldots, v_c$ as follows:

$$f_\theta(x) = \sum_{i=1}^{N_c} v_i \sigma(g_{\theta,i}^{(c)}(x)).$$

We indeed see that the derivative with respect to $g_{\theta,i}^{(c)}(x)$ is $(\sigma'(g_{\theta,i}^{(c)}(x))v_i$, for $i = 1, \ldots, N_c$. □

Then, for k going from $c-1$ to 1 we define (by induction) the vector $\eta_\theta^{(k)}$ of dimension $1 \times N_k$ by

$$\eta_\theta^{(k)} = \underbrace{\eta_\theta^{(k+1)}}_{1\times N_{k+1}} \underbrace{W^{(k+1)}}_{N_{k+1}\times N_k} \underbrace{D_{\sigma'}(g_\theta^{(k)}(x))}_{N_k \times N_k}$$

with

$$W^{(k+1)} := \begin{pmatrix} w_1^{(k+1)} \\ \vdots \\ w_{N_{k+1}}^{(k+1)} \end{pmatrix}$$

in dimension $N_{k+1} \times N_k$, where

$g_\theta^{(k)}(x)$ is the vector of size N_k composed by the values at layer k **just before** the activations σ,

and where for a vector $z = (z_1, \ldots, z_q)$,

$$D_{\sigma'}(z) = \begin{pmatrix} \sigma'(z_1) & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \sigma'(z_q). \end{pmatrix}$$

in dimension $q \times q$.

Lemma 6.2. *For $k = 1, \ldots, c-1$, $\eta_\theta^{(k)}$ is the (line) gradient of the network output with respect to the vector of values at layer k **just before** the activations σ.*

For Lemma 6.2, we make the same comment as stated after Lemma 6.1.

Proof of Lemma 6.2

Assume that the values of the network at layer k just before σ are $g_\theta^{(k)}(x) + z$ for a small $z = (z_1, \ldots, z_{N_k})$. The corresponding output of the network, that we write $f_{\theta,z}(x)$, is a function of $g_\theta^{(k)}(x) + z$ and the parameters of layers $k+1, \ldots, c$ as follows:

$$f_{\theta,z}(x) = h_\theta^{(k+1)} \left\{ W^{(k+1)} \sigma(g_\theta^{(k)}(x) + z) + b^{(k+1)} \right\},$$

where $b^{(k+1)} = (b_1^{(k+1)}, \dots, b_{N_{k+1}}^{(k+1)})^\top$, where for a vector $t = (t_1, \dots, t_q)$ we let $\sigma(t) = (\sigma(t_1), \dots, \sigma(t_q))$ and where $h_\theta^{(k+1)}$ is the function providing the ouput of the network from the values at layer $k+1$ just before the activations σ. We compute a Taylor expansion as $z \to 0$:

$$\begin{aligned} f_{\theta,z}(x) =& h_\theta^{(k+1)} \left\{ W^{(k+1)} \sigma(g_\theta^{(k)}(x)) + b^{(k+1)} + W^{(k+1)} D_{\sigma'}(g_\theta^{(k)}(x)) z + o(\|z\|) \right\} \\ =& h_\theta^{(k+1)} \left\{ W^{(k+1)} \sigma(g_\theta^{(k)}(x)) + b^{(k+1)} \right\} + \eta_\theta^{(k+1)} W^{(k+1)} D_{\sigma'}(g_\theta^{(k)}(x)) z + o(\|z\|) \\ =& f_\theta(x) + \eta_\theta^{(k+1)} W^{(k+1)} D_{\sigma'}(g_\theta^{(k)}(x)) z + o(\|z\|), \end{aligned}$$

because by definition $\eta_\theta^{(k+1)}$ is the gradient vector of $h_\theta^{(k+1)}(\cdot)$ with respect to the input "$\cdot$".

Hence by definition of the gradient vector, the gradient of $f_\theta(x)$ with respect to values of the network at layer k just before σ is

$$\eta_\theta^{(k+1)} W^{(k+1)} D_{\sigma'}(g_\theta^{(k)}(x))$$

which is the definition of $\eta_\theta^{(k)}$. □

Hence backpropagation consists in computing (in this order):

$$\eta_\theta^{(c)} \longrightarrow \cdots \longrightarrow \eta_\theta^{(1)}.$$

Note that we can compute before, in a forward pass (coinciding with computing the output $f_\theta(x)$)

$$g_\theta^{(1)}(x) \longrightarrow \cdots \longrightarrow g_\theta^{(c)}(x).$$

Proposition 6.3. *For $i = 1, \dots, N_c$,*

$$\frac{\partial f_\theta(x)}{\partial v_i} = \sigma(g_{\theta,i}^{(c)}(x)). \tag{6.1}$$

For $k = 1, \dots, c$ and $i = 1, \dots, N_k$,

$$\frac{\partial f_\theta(x)}{\partial b_i^{(k)}} = \eta_{\theta,i}^{(k)}. \tag{6.2}$$

For $k = 2, \dots, c$, $i = 1, \dots, N_k$ and $j = 1, \dots, N_{k-1}$,

$$\frac{\partial f_\theta(x)}{\partial w_{i,j}^{(k)}} = \sigma(g_{\theta,j}^{(k-1)}(x)) \eta_{\theta,i}^{(k)}. \tag{6.3}$$

For $i = 1, \dots, N_1$ and $j = 1, \dots, d$,

$$\frac{\partial f_\theta(x)}{\partial w_{i,j}^{(1)}} = x_j \eta_{\theta,i}^{(1)}. \tag{6.4}$$

Proof of Proposition 6.3

Proof of (6.1)

We can write

$$f_\theta(x) = \sum_{i=1}^{N_c} \sigma(g_{\theta,i}^{(c)}(x)) v_i$$

and $\sigma(g_{\theta,i}^{(c)}(x))$ does not depend on $v_1, \ldots, v_{N_c}$. Hence (6.1) holds.

Proof of (6.2)

We can write, if the scalar $b_i^{(k)}$ is replaced by $b_i^{(k)} + z$ for a small z, defining $f_{\theta,z}(x)$ as the new network output,

$$f_{\theta,z}(x) = h_\theta^{(k)}\left(g_\theta^{(k)}(x) + z e_i^{(N_k)}\right),$$

where $e_i^{(N_k)}$ is the i-th base vector in $\mathbb{R}^{N_k}$ and where $h_\theta^{(k)}$ is the function that maps

the values at layer k just before the activations σ

to

the output of the network.

By Lemma 6.1 or Lemma 6.2, the gradient of $h_\theta^{(k)}(\cdot)$ with respect to "$\cdot$" at $g_\theta^{(k)}(x)$ is $\eta_\theta^{(k)}$. Hence, by a Taylor expansion

$$f_{\theta,z}(x) = h_\theta^{(k)}\left(g_\theta^{(k)}(x)\right) + \eta_{\theta,i}^{(k)} z + o(\|z\|) = f_\theta(x) + \eta_{\theta,i}^{(k)} z + o(\|z\|)$$

and thus (6.2) holds.

Proof of (6.3)

We can write, if the scalar $w_{i,j}^{(k)}$ is replaced by $w_{i,j}^{(k)} + z$ for a small z, defining $f_{\theta,z}(x)$ as the new network output,

$$f_{\theta,z}(x) = h_\theta^{(k)}\left(g_\theta^{(k)}(x) + z\sigma(g_{\theta,j}^{(k-1)}(x)) e_i^{(N_k)}\right).$$

Hence, by a similar Taylor expansion as before

$$f_{\theta,z}(x) = h_\theta^{(k)}\left(g_\theta^{(k)}(x)\right) + \eta_{\theta,i}^{(k)} \sigma(g_{\theta,j}^{(k-1)}(x)) z + o(\|z\|) = f_\theta(x) + \sigma(g_{\theta,j}^{(k-1)}(x)) \eta_{\theta,i}^{(k)} z + o(\|z\|)$$

and thus (6.3) holds.

Proof of (6.4)

We can write, if the scalar $w_{i,j}^{(1)}$ is replaced by $w_{i,j}^{(1)} + z$ for a small z, defining $f_{\theta,z}(x)$ as the new network output,

$$f_{\theta,z}(x) = h_\theta^{(1)}\left(g_\theta^{(1)}(x) + z x_j e_i^{(N_1)}\right).$$

Hence, by a similar Taylor expansion as before

$$f_{\theta,z}(x) = h_\theta^{(1)}\left(g_\theta^{(1)}(x)\right) + \eta_{\theta,i}^{(1)} x_j z + o(\|z\|) = f_\theta(x) + x_j \eta_{\theta,i}^{(1)} z + o(\|z\|)$$

and thus (6.4) holds. □

An example

We let σ be the indentity function for simplicity. We consider a neural network with $c = 2$, $N_0 = d = 2$, $N_1 = 3$, $N_2 = 2$ and with parameters as follows.

θ	value
$w^{(1)}_{1,1}$	1
$w^{(1)}_{1,2}$	-1
$w^{(1)}_{2,1}$	0
$w^{(1)}_{2,2}$	1
$w^{(1)}_{3,1}$	2
$w^{(1)}_{3,2}$	-2
$b^{(1)}_{1}$	0
$b^{(1)}_{2}$	1
$b^{(1)}_{3}$	-1
$w^{(2)}_{1,1}$	1
$w^{(2)}_{1,2}$	1
$w^{(2)}_{1,3}$	-1
$w^{(2)}_{2,1}$	2
$w^{(2)}_{2,2}$	1
$w^{(2)}_{2,3}$	1
$b^{(2)}_{1}$	0
$b^{(2)}_{2}$	1
v_1	1
v_2	1

The input is $x = (1, 2)$. The execution of the forward pass (computing the network output) yields

$$x = \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \quad g^{(1)}_\theta(x) = \begin{pmatrix} -1 \\ 3 \\ -3 \end{pmatrix}, \quad g^{(2)}_\theta(x) = \begin{pmatrix} 5 \\ -1 \end{pmatrix}, \quad f_\theta(x) = 4.$$

The execution of the backward pass (retropropagation yields)

$$\eta^{(2)}_\theta = \begin{pmatrix} \sigma'(g^{(2)}_{\theta,1}(x))v_1 & \cdots & \sigma'(g^{(2)}_{\theta,N_c}(x))v_{N_c} \end{pmatrix} = \begin{pmatrix} v_1 & v_2 \end{pmatrix} = \begin{pmatrix} 1 & 1 \end{pmatrix}$$

and then

$$\eta^{(1)}_\theta = \eta^{(2)}_\theta W^{(2)} D_{\sigma'}(g^{(1)}_\theta(x)) = \begin{pmatrix} 1 & 1 \end{pmatrix} \times \begin{pmatrix} 1 & 1 & -1 \\ 2 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 3 & 2 & 0 \end{pmatrix}.$$

Hence from Proposition 6.3 we have all the partial derivatives as follows.

θ	corresponding partial derivative
$w_{1,1}^{(1)}$	3
$w_{1,2}^{(1)}$	6
$w_{2,1}^{(1)}$	2
$w_{2,2}^{(1)}$	4
$w_{3,1}^{(1)}$	0
$w_{3,2}^{(1)}$	0
$b_1^{(1)}$	3
$b_2^{(1)}$	2
$b_3^{(1)}$	0
$w_{1,1}^{(2)}$	-1
$w_{1,2}^{(2)}$	3
$w_{1,3}^{(2)}$	-3
$w_{2,1}^{(2)}$	-1
$w_{2,2}^{(2)}$	3
$w_{2,3}^{(2)}$	-3
$b_1^{(2)}$	1
$b_2^{(2)}$	1
v_1	5
v_2	-1

We conclude by computing some derivatives "by hand" in order to confirm that backpropagation provides the correct derivative values.

We have

$$\begin{aligned} f_\theta(x) =& v_1 \left\{ w_{1,1}^{(2)} \left[w_{1,1}^{(1)} + 2w_{1,2}^{(1)} + b_1^{(1)} \right] + w_{1,2}^{(2)} \left[w_{2,1}^{(1)} + 2w_{2,2}^{(1)} + b_2^{(1)} \right] + w_{1,3}^{(2)} \left[w_{3,1}^{(1)} + 2w_{3,2}^{(1)} + b_3^{(1)} \right] + b_1^{(2)} \right\} \\ &+ v_2 \left\{ w_{2,1}^{(2)} \left[w_{1,1}^{(1)} + 2w_{1,2}^{(1)} + b_1^{(1)} \right] + w_{2,2}^{(2)} \left[w_{2,1}^{(1)} + 2w_{2,2}^{(1)} + b_2^{(1)} \right] + w_{2,3}^{(2)} \left[w_{3,1}^{(1)} + 2w_{3,2}^{(1)} + b_3^{(1)} \right] + b_2^{(2)} \right\}. \end{aligned}$$

Let us differentiate with respect to $w_{3,2}^{(1)}$. The terms that depend on $w_{3,2}^{(1)}$ are

$$2w_{3,2}^{(1)} w_{1,3}^{(2)} v_1 + 2w_{3,2}^{(1)} w_{2,3}^{(2)} v_2.$$

Differentiating yields

$$2w_{1,3}^{(2)} v_1 + 2w_{2,3}^{(2)} v_2 = 2.(-1).1 + 2.1.1 = 0,$$

confirming the result from backpropagation.

Let us differentiate with respect to $b_1^{(1)}$. The terms that depend on $b_1^{(1)}$ are

$$w_{1,1}^{(2)} b_1^{(1)} v_1 + w_{2,1}^{(2)} b_1^{(1)} v_2.$$

Differentiating yields

$$w_{1,1}^{(2)} v_1 + w_{2,1}^{(2)} v_2 = 1.1 + 2.1 = 3,$$

confirming the result from backpropagation.

As a last example, let us differentiate with respect to $w_{2,3}^{(2)}$. The terms that depend on $w_{2,3}^{(2)}$ are

$$w_{2,3}^{(2)} \left(w_{3,1}^{(1)} + 2w_{3,2}^{(1)} + b_3^{(1)} \right) v_2.$$

Differentiating yields

$$\left(w_{3,1}^{(1)} + 2w_{3,2}^{(1)} + b_3^{(1)}\right) v_2 = (2 + 2.(-2) - 1).1 = -3,$$

confirming the result from backpropagation.

Bibliography

Martin Anthony and Peter L Bartlett. *Neural network learning: Theoretical foundations*. cambridge university press, 2009.

Peter L Bartlett, Nick Harvey, Christopher Liaw, and Abbas Mehrabian. Nearly-tight VC-dimension and pseudodimension bounds for piecewise linear neural networks. *Journal of Machine Learning Research*, 20(63):1–17, 2019.

George Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314, 1989.

Walter Rudin. *Real and Complex Analysis*. Mc Graw Hill, 1998.

EXERCICES