Topics in Math of ML, Lecture 6
Neural Networks (I)

Lénaïc Chizat*

October 27, 2025

1 Introduction

So far in the class, we have considered prediction with *linearly-parameterized* predictors

$$f_\theta(x) = \langle \theta, \phi(x) \rangle_{\mathcal{H}}$$

with a given, fixed, feature map $\phi : \mathcal{X} \to \mathcal{H}$ (either explicit, implicit via a kernel $k(x, x')$, or approximated with random features). Now we will consider models that are non-linear both in x and θ: this is the realm of (Artificial) Neural Networks (NNs). The program of today is:

- presentation of neural networks and vocabulary
- infinite width perspective, norms and approximation results

2 Definitions of Neural Networks

Historically, NNs have first appeared in the form of *fully-connected neural networks*, a.k.a. *multi-layer percetrons* (MLP):

$$\begin{aligned} f_\theta(x) &= W_L \sigma(\dots \sigma(W_2 \sigma(W_1 x + b_1) + b_2) \dots) + b_L, \\ \theta &= (W_1, \dots, W_L, b_1, \dots, b_L) \in \mathbb{R}^{m \times d} \times \mathbb{R}^{m \times m} \times \cdots \times \mathbb{R}^{1 \times m} \times \mathbb{R}^m \times \cdots \times \mathbb{R}^1 \end{aligned}$$

- L is called the *depth* or *number of layers* of the NN;
- m is the *width* of the neural network
- W_ℓ are the *weight matrices*; b_j are often called the "biases" by practitioners, but this leads to an unfortunate conflict with the statistical notion of bias, so we prefer to call them *intercepts*.
- $\sigma : \mathbb{R} \to \mathbb{R}$ is an *activation function* (a.k.a. non-linearity) which acts entrywise on vectors. It is typically one of the following:
  - sigmoid $\sigma(u) = \frac{1}{1+e^{-u}}$
  - step $\sigma(u) = 1_{u>0}$
  - rectified linear unit (ReLU): $\sigma(u) = (u)_+ = \max\{0, u\}$ (or its powers, such as square ReLU);

*EPFL lenaic.chizat@epfl.ch

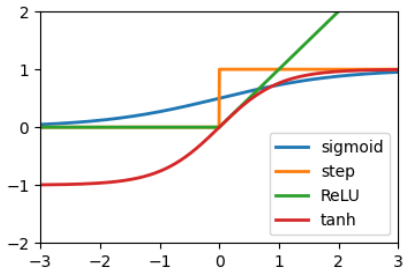

Figure 1: Common activation functions

  - hyperbolic tangent $\sigma(u) = \tanh(u) = \frac{e^u - e^{-u}}{e^u + e^{-u}}$

- there is an analogy between this architectures and neurons in the brain; but this is not a complete analogy, and this is not a justification for the performance of these methods.

In practice other *architectures* are used such as *residual networks* (ResNets), *convolutional networks* (ConvNets), *attention layers* (in the Transformer architecture), etc. They are all built – with important differences – on the basic architecture above. With the introduction of those more and more complicated architectures, the notion of NN has gradually become more general. Today, an appropriate definition of a NN would be:

> *Neural Net : Any function f_θ that can be described by a computer program which admits a (notion of) differential with respect to its parameters θ.*

Learning with such functions is called *deep learning* or *differentiable programming*. In today's course we will focus on the simplest versions of NNs that can be analyzed theoretically, which is the two-layer perceptron.

Optimization The parameters θ of the NN are adjusted by running stochastic gradient descent (SGD) (or a variant such as the algorithm known as "Adam") on the (parameterized) empirical risk

$$\begin{aligned} F(\theta) &= \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, f_\theta(x_i)) + \frac{\lambda}{2}\|\theta\|_2^2 \\ &= \frac{1}{n}\sum_{i=1}^{n} F_i(\theta) \end{aligned}$$

where $\lambda \geq 0$ is an optional weight regularization term (called *weight decay* in this context; the reason for this name is that this corresponds to contracting the weights by a factor $(1 - \eta\lambda)$ at each iteration of Algorithm 1).

In contrast to the case of linearly-parameterized predictor, this is a *non-convex* function of the parameters θ and thus there is in general no guarantee that the optimization algorithm converges to a minimizer. We'll present this algorithm and discuss its behavior in future lectures.

See https://playground.tensorflow.org/ for an interactive vizualization of the training process of MLPs.

Algorithm 1: Stochastic Gradient Descent (SGD)

- Hyper-parameters: $\eta > 0$ step-size (a.k.a. learning rate), $\mu_0 \in \mathcal{P}(\mathbb{R}^p)$ initial distribution
- Sample $\theta_0 \sim \mu_0$
- for $t = 1, 2, \ldots$
  - Sample $i_t \in \{1, \ldots, n\}$
  - $\theta(t+1) = \theta(t) - \eta \nabla F_{i_t}(\theta(t))$

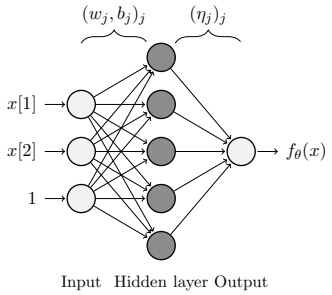

Figure 2: Graph representation of a two-layer NN

3 Two layer NNs and their infinite width limit

Today we will focus on two-layer neural networks (2NNs). Consider $\mathcal{X} = \mathbb{R}^d$ and the set of functions that can be written as

$$f_\theta(x) = \sum_{j=1}^{m} \eta_j \sigma(w_j^\top x + b_j), \qquad \theta = (w_j, b_j, \eta_j)_{j=1}^m \in (\mathbb{R}^d \times \mathbb{R} \times \mathbb{R})^m$$

Note that the intercepts b_j can be removed if one considers inputs x with 1 as their last coordinate. In this architecture, w_j are the called the *input weights* and η_j the *output weights*.

3.1 Link with kernel methods and RKHS norm $\mathcal{F}_2$

A 2NN corresponds to a linear predictor with feature vector of dimension m

$$\phi(x)_j = \frac{1}{\sqrt{m}} \sigma(w_j^\top x + b_j)$$

parameterized by all weights w_j, b_j and with corresponding kernel

$$\hat{k}_m(x, x') = \frac{1}{m} \sum_{i=1}^{m} \sigma(w_j^\top x + b_j) \sigma(w_j^\top x' + b_j).$$

In general, this kernel itself depends on the data because the parameters $(w_j, b_j)_{j=1}^m$ are "trained". From a statistical viewpoint, nothing much can be deduced from this point of view.

Now if one runs a slightly modified version of Algorithm 1 where the input weights $(w_j, b_j)_{j=1}^m$ are kept fixed after their random initialization (and therefore do not depend on the data), this corresponds to solving ERM with the kernel $\hat{k}_m$ (assuming $\lambda > 0$). With random iid input weights (w_j, b_j) for $j = 1, \ldots, m$, we get in the *infinite width limit* ($m \to \infty$), by the law of large numbers

$$\hat{k}_m(x, x') \to k(x, x') = \mathbf{E}[\sigma(w_j^\top x + b_j) \sigma(w_j^\top x' + b_j)].$$

This kernel k is called the *conjugate* kernel of the NN[1]. For some activation functions, this kernel has an explicit form (for instance "arccosine kernels", see exercices).

In this context (i.e. when the input weights are untrained and random), two-layer perceptrons are thus exactly doing kernel ridge regression with a random feature approximation. We will see later another (more subtle) connection with kernel methods for fully trained wide NNs for certain choices of initialization (the theory of Neural Tangent Kernel (NTK)).

Expression of the norm Let $\tau \in \mathcal{P}(\mathbb{R}^d \times \mathbb{R})$ be the law of (w_j, b_j). The kernel k can be represented by the feature map $\phi : \mathcal{X} \to L^2(\tau)$ given by

$$\phi_{w,b}(x) = \sigma(w^\top x + b)$$

and the RKHS norm is thus

$$\|f\|_{\mathcal{F}_2} = \min_\eta \left\{ \|\eta\|_{L_2(\tau)} \; ; \; f(x) = \int_{\mathbb{R}^d \times \mathbb{R}} \eta(w,b)\sigma(w^\top x + b)\mathrm{d}\tau(w,b), \forall x \in \mathcal{X} \right\}.$$

- Note that the minimum is attained because the constraint set is weakly closed in $L^2(\mathrm{d}\tau)$ (when non-empty) and the objective is weakly lower semi-continuous, with weakly compact sublevel sets.
- when σ is positively homogeneous, this kernel belongs to the class of dot-product kernels, which can be analyzed similarly to translation-invariant kernels (except that the analysis involves spherical harmonics instead of Fourier harmonics). Their behavior is "comparable" to Matern kernels.

In general the input weights are also optimized over and the neural network is thus also doing *feature/representation* learning. This more complex behavior leads to better performances in practice, and the theory of NN is mostly concerned about this phenomenon.

3.2 Variation norm $\mathcal{F}_1$

Another function space that is more faithful of representation capabilities of *fully trained* NN (see justification in the paragraph "finite width NNs) can be defined by replacing formally the L_2 norm by a L_1 norm:

$$\inf_\eta \left\{ \int |\eta(w,b)| \mathrm{d}\tau(w,b) \; ; \; f(x) = \int_{\mathbb{R}^d \times \mathbb{R}} \eta(w,b)\sigma(w^\top + b)\mathrm{d}\tau(w,b), \forall x \in \mathcal{X} \right\}.$$

The above infimum is not always attained and the above problem is better formulated in the space of measures. Given a constraint set $S \subset \mathbb{R}^d \times \mathbb{R}$, we define the $\mathcal{F}_1$ norm as follows:

$$\|f\|_{\mathcal{F}_1} \coloneqq \min_\mu \Big\{ \underbrace{\int |\mathrm{d}\mu(w,b)|}_{\text{Total variation norm of } \mu} \; ; \; \mu \in \mathcal{M}(S) \; ; \; f(x) = \int_{\mathbb{R}^d \times \mathbb{R}} \sigma(w^\top x + b)\mathrm{d}\mu(w,b), \forall x \in \mathcal{X} \Big\}.$$

where $\mathcal{M}(S)$ is the space of finite, signed Borel measures on S. The set S would correspond to the support of τ in the previous display. The resulting norm is not Hilbertian anymore, but is a Banach space norm. The min is attained, provided S is compact (then sublevels are compact and the objective is lower-semicontinuous, for the weak topology on $\mathcal{M}(S)$).

Let us call $\mathcal{F}_i$ the space of functions with finite $\|\cdot\|_{\mathcal{F}_i}$ norm.

[1] Note that in general the conjugate kernel depends on both the choice of architecture and of random distribution of weights. It is defined as the kernel associated with the feature representation after the last layer of nonlinearity, at random initialization, and in the infinite width limit.

Remark 3.1. *By Jensen's inequality for any $\eta \in L^2(\tau)$ it holds,*

$$\Big(\int |\eta(w,b)| \mathrm{d}\tau(w,b) \Big)^2 \leq \int |\eta(w,b)|^2 \mathrm{d}\tau(w,b).$$

Hence one has $\|f\|_{\mathcal{F}_1} \leq \|f\|_{\mathcal{F}_2}$ hence $\mathcal{F}_2 \subset \mathcal{F}_1$.

Goals. We now describe the approximation property of $\mathcal{F}_1$ for functions defined $B_R = \{x \in \mathbb{R}^d; \|x\|_2^2 \leq R\}$ for the specific case of the ReLU activation function $\sigma(u) = \max\{u, 0\} = (u)_+$. We will consider measures supported on the set

$$S = \{(w,b), \|w\|_2 = 1, |b| \leq R\}.$$

Finite width NNs In this context, if $f(x) = \sum_{j=1}^m \eta_j (w_j^\top x + b_j)_+$ for "neurons" such that $(w_j, b_j) \in S$, $\forall j$, then $\|f\|_{\mathcal{F}_1} \leq \|\eta\|_1$ as this corresponds to a representation with the measure $\mu = \sum_{j=1}^m \eta_j \delta_{(w_j,b_j)}$. In contrast, no such guarantee exists for $\|f\|_{\mathcal{F}_2}$ (in fact, it can be shown that $\|f\|_{\mathcal{F}_2} = +\infty$ for $d > 1$ because then $\mathcal{F}_2$ only contains continuously differentiable functions; this is analogous to the regularity property in RKHS associated to translation invariance kernels, see Lecture 5).

4 Approximation in 1D

Assume that $f : [-R, R] \to \mathbb{R}$ is differentiable with an absolutely continuous derivative and that $f(0) = f'(0) = 0$. Then by Taylor expansion with integral remainder for $x \in [-R, R]$:

$$\begin{aligned} f(x) &= \int_0^x (x-b) f''(b) \mathrm{d}b && (1)\\ &= \int_0^R f''(b)(x-b)_+ \mathrm{d}b + \int_{-R}^0 f''(b)(b-x)_+ \mathrm{d}b && (2)\\ &= \int_{-R}^0 f''(-b)(x+b)_+ \mathrm{d}b + \int_{-R}^0 f''(b)(-x+b)_+ \mathrm{d}b && (3) \end{aligned}$$

where the second expression is obtained by separating the cases $x \geq 0$ and $x \leq 0$. This is exactly the representation of g as an infinitely wide ReLU 2NN (in 1D, $\|w\|_2 = 1$ is equivalent to $w \in \{-1, +1\}$ hence the presence of two terms with $+x$ and $-x$ inside the ReLU). It follows

$$\|f\|_{\mathcal{F}_1} \leq \int_{-R}^R |f''(b)| \mathrm{d}b \tag{4}$$

To remove the constraints on $f(0)$ and $f'(0)$ one can notice that for $x \in [-R, R]$,

$$\begin{aligned} f(0) &= \frac{f(0)}{2R} \Big((x+R)_+ + (-x+R)_+ \Big), && (5)\\ x f'(0) &= f'(0)(x)_+ - f'(0)(-x)_+ && (6) \end{aligned}$$

Since in general $f(x) = f(0) + x f'(0) + \int_0^x (x-b) f''(b) \mathrm{d}b$ this leads to

$$\|f\|_{\mathcal{F}_1} \leq \frac{|f(0)|}{R} + 2|f'(0)| + \int_{-R}^R |f''(b)| \mathrm{d}b \tag{7}$$

Note that here the measure over $\{-1,1\}\times[-R,R]$ representing f that we have constructed is

$$\mu = \frac{f(0)}{2R}(\delta_{(1,R)}+\delta_{(-1,R)}) + f'(0)(\delta_{(1,0)}-\delta_{(-1,0)}) \\ + f''(-\cdot)(\delta_1\otimes \mathrm{Leb}|_{[-R,0]}) + f''(\cdot)(\delta_{-1}\otimes \mathrm{Leb}|_{[-R,0]}) \quad (8)$$

and the upper bound on $\|f\|_{\mathcal{F}_1}$ is just the total variation norm of μ.

Remark 4.1. *With slightly more work, this bound can be extended to the case of functions which are not differentiable at* 0*, see [Bach, 2022, Eq. (9.2)].*

5 Approximation in higher dimension

Let us assume that $f:\mathbb{R}^d\to\mathbb{R}$ is continuous on the centered ball of radius R. Then we can write

$$f(x) = \frac{1}{(2\pi)^d}\int_{\mathbb{R}^d}\hat{f}(\omega)e^{i\omega^\top x}\mathrm{d}\omega, \qquad \hat{f}(\omega) \coloneqq \int_{\mathbb{R}^d} f(x)e^{-i\omega^\top x}\mathrm{d}x. \quad (9)$$

Taking the real part in the integrand we get

$$f(x) = \frac{1}{(2\pi)^d}\left(\int_{\mathbb{R}^d}\Re(\hat{f}(\omega))\cos(\omega^\top x)\mathrm{d}\omega - \int_{\mathbb{R}^d}\Im(\hat{f}(\omega))\sin(\omega^\top x)\mathrm{d}\omega\right), \quad (10)$$

By the subadditivity property of norms, we have

$$\|f\|_{\mathcal{F}_1} \le \frac{2}{(2\pi)^d}\int_{\mathbb{R}^d}|\hat{f}(\omega)|N(\omega)\mathrm{d}\omega, \qquad N(\omega)\coloneqq\max\{\|\cos(\omega^\top\cdot)\|_{\mathcal{F}_1},\|\sin(\omega^\top\cdot)\|_{\mathcal{F}_1}\}. \quad (11)$$

(There are various ways to extend f outside of the ball of radius R, which lead to different $\hat{f}$: we can minimize over these extensions to get tighter upper bounds).

We can explicitly build an (infinite width) neural network that represents $\cos(\omega^\top\cdot)$ by taking the definition of μ in Eq. (8) and replacing $\{-1,1\}$ by the points in the unit sphere $\{-\omega/\|\omega\|_2,\omega/\|\omega\|_2\}$ (note that this step works in the context of the $\mathcal{F}_1$ norm, but not of the $\mathcal{F}_2$ norm!). By the previous section (Eq. (7)), it thus follows

$$N(\omega) \le \frac{1}{R}+2\|\omega\|_2+2R\|\omega\|_2^2 \le \frac{1}{R}(1+\sqrt{2}R\|\omega\|_2)^2 \le \frac{2}{R}(1+2R^2\|\omega\|_2^2)$$

where we have used that $(a+b)^2\le 2(a^2+b^2)$. Thus we obtain

$$\|f\|_{\mathcal{F}_1} \le \frac{1}{(2\pi)^d}\frac{4}{R}\int|\hat{f}(\omega)|(1+2R^2\|\omega\|_2^2)\mathrm{d}\omega, \quad (12)$$

which can be interpreted as a measure of smoothness of f; but notice the absence of a square on $|\hat{f}(\omega)|$ which makes it possible for the spectrum of f to "concentrate" (e.g. on a smaller dimensional subspace) without having the norm exploding. This feature distinguishes $\mathcal{F}_1$ from $\mathcal{F}_2$.

Link with Sobolev regularity For $s>d/2$ we have

$$\begin{aligned}\|f\|_{\mathcal{F}_1} &\le \frac{1}{(2\pi)^d}\frac{4}{R}\int|\hat{f}(\omega)|(1+2R^2\|\omega\|_2^2)^{1+s/2}(1+2R^2\|\omega\|_2^2)^{-s/2}\mathrm{d}\omega \\ &\le \frac{1}{(2\pi)^d}\frac{4}{R}\sqrt{\int|\hat{f}(\omega)|^2(1+2R^2\|\omega\|_2^2)^{2+s}\mathrm{d}\omega}\sqrt{\int(1+2R^2\|\omega\|_2^2)^{-s}\mathrm{d}\omega} \\ &\lesssim C\|f\|_{H^{2+s}}\end{aligned}$$

where H^{2+s} is the Sobolev space of order $2 + s > 2 + d/2$ (the constraint $s > d/2$ is chosen so that the rightmost integral is finite). Thus the approximation properties of such Sobolev spaces apply, similarly to the case of kernel methods with translation invariant kernels. However, we will see in the next that $\mathcal{F}_1$ has in fact much richer than such RKHS.

References

Francis Bach. Learning theory from first principles. *Draft of a book, version of Sept.*, 6:2022, 2022.