Deep learning, notes

L. Borasi

September 25, 2023

Abstract

Personal notes while reading the book [GBC16].

Contents

1 MLP

1.1 General notation

Let the output of a NN be a vector $f_{\text{out}} \in \mathcal{H}_{\text{out}}$ where $\mathcal{H}_{\text{out}}$ is a Hilbert space with scalar product $\langle \cdot, \cdot \rangle$ and Hilbert norm $\|f\| \stackrel{\text{def}}{=} \sqrt{\langle f, f \rangle}$.

Consider $(e_i)_{i \in I}$ to be a Hilbert basis of $\mathcal{H}_{\text{out}}$. Then

$$\|f_{\text{out}}\|^2 = \sum_{i \in I} \langle f_{\text{out}}, e_i \rangle^2 = \sum_{i \in I} [f_{\text{out}}]_i^2 \quad v \in \mathcal{H}_{\text{out}},$$

where we have used the notation $[f]_i \stackrel{\text{def}}{=} \langle f, e_i \rangle$, $i \in I$, to denote the i-th component of f.

Similarly let $\mathcal{H}_{\text{in}}$ the vector space of inputs. Moreover, let $\mathbb{V}$ be the vector space of weights. Following the standard convention we consider a weight $V \in \mathbb{V}$ to be composed of two components: $V = (b, w)$ where $b \in \mathbb{B}$, $w \in \mathbb{W}$, with $\mathbb{V} = \mathbb{B} \oplus \mathbb{V}$. An element $W \in \mathbb{W}$ is a collection of weighted edges which characterize the NN. An element $B \in \mathbb{B}$ is the collection of "additive coefficients" (see below).

We now consider the *transition function* $F : \mathcal{H}_{\text{in}} \times \mathbb{V} \to \mathcal{H}_{\text{out}}$. This function characterizes the NN completely, in the sense that, given the weights $w \in \mathcal{W}$ and the inputs f_{in}, $F(f_{\text{in}}, w) \in \mathcal{H}_{\text{out}}$ is the output returned by the NN.

We consider the *cost function* $S : \mathbb{V} \to \mathbb{R}$ given by

$$S(V) \stackrel{\text{def}}{=} \sum_{f_{\text{in}} \in \mathcal{H}_{\text{in}}} \|F(f_{\text{in}}, V) - F_\infty(f_{\text{in}})\|^2, \quad V \in \mathbb{V}.$$

Here $F_\infty : \mathcal{H}_{\text{in}} \to \mathcal{H}_{\text{out}}$ is the "reference function", that is a function that returns the "reference output" for each input (in principle we could have absorbed F_∞ into F, but for clarity we keep them separated).

1.2 Gradient descent

Theorem 1.1. *For $U \subset \mathcal{W}$ compact, and $w \in U$ fixed, there exists an $\epsilon > 0$ such that*

$$S(w - \epsilon \nabla S(w)) \leq S(w).$$

and if $\nabla S(w) \neq 0$, then $\leq$ can be replaced by $<$.

Proof sketch. For a fixed $V \in \mathbb{V}$, consider $\nabla S(V)$ as an element of $\mathbb{V}$. Consider the expansion

$$S(V - \epsilon \nabla S(V)) = S(V) - \epsilon \|\nabla S(V)\|^2 + R(V, \epsilon) \quad V \in \mathcal{U} \subset \mathbb{V}, \quad 0 < \epsilon < 1.$$

Let $\mathcal{U} \subset \mathbb{V}$, and $0 < \epsilon < 1$, such that $R(w, \epsilon) \leq C(U)\epsilon^2$, $w \in U$. Then

$$\begin{aligned} S(V - \epsilon \nabla S(V)) &= S(V) - \epsilon \|\nabla S(V)\|^2 + \epsilon^2 C(\mathcal{U}) \\ S(V - \epsilon \nabla S(V)) - S(V) &= -\epsilon \|\nabla S(V)\|^2 + \epsilon^2 C(\mathcal{U}) \\ &\leq -\epsilon \|\nabla S(V)\|^2 + \epsilon^2 |C(\mathcal{U})|. \end{aligned}$$

Assume $\|\nabla S(V)\|_{\mathbb{V}}\| \neq 0$, then let ϵ such that $\epsilon |C(\mathcal{U})| < \frac{1}{2} \|\nabla S(w)\|_{\mathcal{W}}$. Then

$$S(V - \epsilon \nabla S(V)) - S(V) < -\frac{\epsilon}{2} \|\nabla S(V)\|_{\mathbb{V}} < 0.$$

This concludes the proof. □

1.3 MLP and filtration

We consider a MLP with $N \in \mathbb{N}$ neurons per layer and $L \in \mathbb{N}$ layers. Let, with reference to the notation of subsection 1.1, $\mathcal{H}^{\text{in}} \stackrel{\text{def}}{=} \mathbb{R}^N$, $\mathbb{B} \stackrel{\text{def}}{=} \mathbb{R}^N \times \mathbb{R}^L$, $\mathbb{W} \stackrel{\text{def}}{=} \mathbb{R}^N \times \mathbb{R}^N \times \mathbb{R}^L$. Moreover we consider each layer as being the input of the next. Therefore we set $\mathcal{H}$ to be a copy of $\mathcal{H}^{\text{in}}$ as the abstract space of inputs to a generic layer ℓ, with $\ell \in \{0, \dots, L\}$. Here the layer with $\ell = 0$ represents the input layer, whereas the layer with $\ell = L$ represents the output layer. So the MLP has actually $L + 1$ layers and $L - 1$ hidden layers.

We consider the following decomposition of the weight over the layers. By this we mean the following. First, we decompose $\mathbb{V} = \mathbb{B} \oplus \mathbb{V}$ as a direct sum of spaces indexed by the layers:

$$\mathbb{V} = \underbrace{\mathcal{V} \oplus \cdots \oplus \mathcal{V}}_{L \text{ times}}, \quad \mathcal{V} = \mathcal{B} \oplus \mathcal{W}, \quad \mathcal{B} \cong \mathbb{R}^N, \quad \mathcal{W} \cong \mathbb{R}^N \times \mathbb{R}^N, \tag{1}$$

Now we have two equivalent options to think of the weights in this decomposition. Consider a layer ℓ. Then this layer will have depend on a set of inputs and a set of weights. The set of weights that contribute to the layer ℓ, but no layer ℓ' with $\ell' < \ell$, is by definition isomorphic to $\mathcal{V}$. We can think of this set as either a set of "additional inputs" for the layer ℓ or as a set describing the "state" of layer ℓ. The first interpretation is in a sense reminiscence of the von Neumann architecture, where data= inputs and instructions=weights have the same representation (on the memory); on the other hand the second interpretation is reminiscent of the Harvard architecture, where data are instructions have different representation. In [GBC16] the second option is taken. To have some diversity, we take the first. Beside the philosophical aspect, the only effective difference is wether we count the weights from 0 to $L - 1$ or from 1 to L (cf. (2) below). Therefore, we order the $\mathcal{V}$ in the decomposition (1) starting from 0 and ending with $L - 1$. The ℓ-th $\mathcal{V}$ in the decomposition will be thought of as an additional input to the layer $\ell + 1$.

The decomposition (1) gives rise to a "filtration" of spaces of weights. We define the spaces:

$$\begin{aligned} \mathbb{V}_{[\![\ell+h,\ell]\!]} &\stackrel{\text{def}}{=} \bigoplus_{\ell=\ell}^{\ell+h} \mathcal{V} = \underbrace{\mathcal{V} \oplus \cdots \oplus \mathcal{V}}_{h \text{ times}}, \quad h \in \{0, \dots, L - 1 - \ell\}, \quad \ell \in \{0, \dots, L - 1\}, \\ \mathbb{V}_{\ell} &\stackrel{\text{def}}{=} \mathbb{V}_{[\![\ell,\ell]\!]} = \mathcal{V}, \quad \ell \in \{0, \dots, L - 1\},. \end{aligned}$$

These spaces define a filtration, i.e. we have a sequence of spaces $\mathbb{V}_{[\![\ell,0]\!]}$, $\ell \in \{1, \dots, L\}$, which satisfy

$$\mathbb{V} = \mathbb{V}_{[\![L-1,0]\!]} \supset \cdots \supset \mathbb{V}_{[\![1,0]\!]} \supset \mathbb{V}_{[\![0,0]\!]} \cong \mathcal{V}.$$

We employ the same notation for the space $\mathbb{B}$, $\mathbb{W}$. The space $\mathbb{V}_{[\![\ell+h,\ell]\!]}$ represents the set of weight which "connect" the layer ℓ with the layer $\ell + h + 1$. We think of an element $V \in \mathbb{V}$ as an L-tuple:

$$V = (v^{L-1}, v^{L-2}, \dots, v^0), \quad v^\ell \in \mathcal{V}, \quad v^\ell = (b^\ell, w^\ell), \quad b^\ell \in \mathcal{B}, \quad w^\ell \in \mathcal{W}, \quad \ell \in \{0, \dots, L - 1\}.$$

We denote the restriction of $V \in \mathbb{V}$ to $\mathbb{V}_{[\![\ell+h,\ell]\!]}$ by $V_{[\![\ell+h,\ell]\!]}$ with

$$V_{[\![\ell+h,\ell]\!]} = (v_{\ell+h}, v_{\ell+h-2}, \dots, v_\ell), \quad v_\ell \in \mathcal{V}, \quad \ell \in \{\ell, \dots, \ell + h\}.$$

Let $\Phi^\ell : \mathcal{H} \to \mathcal{H}$ be a "vector activation function". To describe a simple MLP, we take Φ of the form

$$[\Phi^\ell(x)]_n = \varphi^\ell(x_n), \quad x = (x_1, \dots, x_N) \in \mathcal{H},$$

where $[\cdot]_n$, $n \in \{1, \dots, N\}$, denotes the n-th component of a vector in $\mathcal{H}$; here $\varphi^\ell : \mathbb{R} \to \mathbb{R}$ is the activation function of the n-th neuron on the ℓ-th layer and every neuron on the same layer has the same activation function.

We define recursively the MLP characteristic function $F : \mathbb{V} \times \mathcal{H}_{\text{in}} \to \mathcal{H}_{\text{out}}$:

$$\begin{aligned} F(V, f) &\overset{\text{def}}{=} x^L, \\ x^{\ell+1} &\overset{\text{def}}{=} b^\ell + w^\ell \Phi^{\ell+1}(x^\ell), \quad \ell \in \{0, \dots, L-1\}, \\ x^0 &= f^{in}, \end{aligned} \tag{2}$$

where $v = (w, b) \in \mathcal{V}$. For example for $N = 1$ and $L = 3$ we have:

$$F(f_{\text{in}}, v) \overset{\text{def}}{=} w^{(2)} \varphi(w^{(1)} \varphi(w^{(0)} f_{\text{in}} + b^{(0)}) + b^{(1)}) + b^{(2)}.$$

We see in (2) that the weights can be thought of as either additional inputs or as state of a layer. In (2) the weights and the inputs appear on the right hand side with the same index ℓ and can be thoughts as data fed to the layer $\ell + 1$. On the other hand we have $\Phi^{\ell+1}$ because we think of Φ^ℓ as the activation function of the layer ℓ. The more common (and perhaps sensible) convention (cf. [GBC16]) is to replace the second line of (2) with

$$x^\ell = b^\ell + w^\ell \Phi^\ell(x^{\ell-1}), \quad \ell \in \{1, \dots, L\},$$

where for clarity we have translated the ℓ to $\ell + 1$. Then the weights (b^ℓ, w^ℓ) would be thought of as part of the function X^ℓ which sends $x^{\ell-1} :\mapsto x^\ell$ (we will come back to this function in (3) below).

We define the cost function to be $S : \mathbb{V} \to \mathbb{R}$.

$$S(v) \overset{\text{def}}{=} \sum_{f^{\text{in}} \in \mathcal{H}^{\text{in}}} \| F(f_{\text{in}}, v) - F_\infty(f_{\text{in}}) \|^2.$$

We want to compute the gradient of $S(V)$. We want to do this rigorously and explicitly so that the implementation of the algorithm becomes straightforward. In (2) each x^ℓ is an element of $\mathcal{H}$ and not a function. To take derivative we need a notation which expresses the dependence of the x^ℓ, $\ell \in \{0, \dots, L\}$ on the weights (and on each other). We do this taking advantage of the "filtration", i.e. on the fact that each layer can be thought as having a transition function which takes as inputs the outputs of the previous layer.

To express this concretely we define the ℓ-th "*layer transition function*":

$$X_v^{\ell+1}(x) \overset{\text{def}}{=} b^\ell + w^\ell \Phi^{\ell+1}(x), \quad v = (b, w) \in \mathcal{V}, \quad x \in \mathcal{H} \quad \ell \in \{0, \dots, L-1\}. \tag{3}$$

We then define the "*filtered transition functions*" to be the functions $F^{\ell', \ell}$, $\ell' > \ell$, given by

$$\begin{aligned} F^{\ell+h+1, \ell}(V_{[\![\ell+h, \ell]\!]}, x) &\overset{\text{def}}{=} X^{\ell+h}_{v^{\ell+h-1}}(\dots X^{\ell+2}_{v^{\ell+1}}(X^{\ell+1}_{v^\ell}(x)) \dots) \\ &\equiv (X^{\ell+h}_{v^{\ell+h-1}} \circ \dots \circ X^{\ell+2}_{v^{\ell+1}} \circ X^{\ell+1}_{v^\ell})(x), \qquad h \in \{0, \dots, L-1-\ell\}, \quad \ell \in \{0, \dots, L-1\}, \end{aligned}$$

where $\circ$ denotes composition of functions. By definition we have

$$F^{L,0}(V, f^{\text{in}}) = F(V, f^{\text{in}}), \quad F^{\ell+1, \ell} \equiv X^{\ell+1}, \quad \ell \in \{0, \dots, L-1\}. \tag{4}$$

Moreover the filtered transition functions satisfy the composition rule

$$F^{L,0}(V, f^{\text{in}}) = F^{L,\ell}(V_{[\![L-1,\ell]\!]}, F^{\ell,0}(V_{[\![\ell-1,0]\!]}, f^{\text{in}})). \tag{5}$$

Finally consider the set of variables $(x^\ell)_{\ell \in \{0, \dots, L\}}$ defined in (2). Each $x^\ell + 1$, $\ell \in \{0, \dots, L-1\}$, is the output after the first ℓ layers have been applied to the inputs, that is, we have

$$x^{\ell+1} = F^{\ell+1,0}(V_{[\![\ell,0]\!]}, f^{\text{in}}), \quad \ell \in \{0, \dots, L-1\}. \tag{6}$$

With this notation, we can rewrite (2) in the following way:

$$\begin{aligned} F(V, f^{\text{in}}) &= F^{L,0}(V, f^{\text{in}}), \\ F^{\ell+1,0}(V_{[\![\ell,0]\!]}, f^{\text{in}}) &= b^\ell + w^\ell \phi^{\ell+1}(F^{\ell,0}(V_{[\![\ell-1,0]\!]}, f^{\text{in}})), \quad \ell \in \{0, \dots, L-1\}. \end{aligned} \tag{7}$$

We further rewrite these relations in a way that makes the parallel with (2) obvious. Note first that

$$F^{\ell+1,0}(V_{[\![\ell,0]\!]}, f^{\text{in}}) = X^{\ell+1}_{v^\ell}(X^{\ell}_{v^{\ell-1}}(x^{\ell-1})),$$
$$F^{\ell,0}(V_{[\![\ell-1,0]\!]}, f^{\text{in}}) = X^{\ell}_{v^{\ell-1}}(x^{\ell-1}),$$

where the $(x^\ell)_{\ell\in\{0,\dots,L\}}$ are defined in (2) and satisfy (6). Then we can rewrite (2) as follows

$$\begin{aligned} F(V, f^{\text{in}}) &= X^{L}_{v^{L-1}}(x^{L-1}),\\ X^{\ell+1}_{v^\ell}(X^{\ell}_{v^{\ell-1}}(x^{\ell-1})) &= b^\ell + w^\ell \Phi^{\ell+1}(X^{\ell}_{v^{\ell-1}}(x^{\ell-1})), \quad \ell \in \{0,\dots,L\}. \end{aligned} \tag{8}$$

Comparing (8) with (2), we see that we have the same form or inductive relation but in (8) the symbols X^ℓ are now functions.

Strong of this (perhaps a bit over complicated..) notation we are ready to compute the gradient ∇S.

1.4 Back-propagation

For convenience we call *gradient* both the gradient of a scalar-valued function and the Jacobian of a vector valued function. We shall use the "chain-rule" for the gradient in the following form (cf. [Boo03, formula (1.4) p. 23, and Theorem (2.3) p. 27]). Given two functions $F : \mathbb{R}^\mu \to \mathbb{R}^\nu$, $G : \mathbb{R}^\nu \to \mathbb{R}^\rho$, $\mu,\nu,\rho \in \mathbb{N}$, we let $H \stackrel{\text{def}}{=} G \circ F : \mathbb{R}^\mu \to \mathbb{R}^\rho$. Then we have

$$(DH)(a) = (DG)(F(a))(DF)(a), \quad a \in \mathbb{R}^\nu, \tag{9}$$

where DF and DG represent the gradients of F and G and where on the right-hand side we have the matrix product of the matrix $(DG)(F(a))$ with the matrix $(DF)(a)$. Note that these two matrices have, in general, different shapes: $DG(F(a))$ is a linear map $\mathbb{R}^\rho \to \mathbb{R}^\nu$, hence it is represented by a matrix of shape (ρ,ν), whereas $DF(a)$ is a linear map $\mathbb{R}^\nu \to \mathbb{R}^\mu$, thus its matrix representation has shape (ν,μ).

We use the same convention as [Boo03] with regard to the definition of the matrix $DF(a)$, that is

$$[DF(a)]_{n_1 n_2} \stackrel{\text{def}}{=} \frac{\partial F_{n_1}(x)}{\partial x_{n_2}}\Big|_{x=a}.$$

Before computing the derivatives we introduce two notational conventions which will make the formulas clearer.

First: Consider a filtered transition function $F^{\ell',\ell} : \mathbb{V}_{[\![\ell',\ell]\!]} \times \mathcal{H} \to \mathcal{H}$, $\ell' > \ell$, $\ell',\ell \in \{0,\dots,L+1\}$. Since $F^{\ell',\ell}$ is a function of a pair of vector-variables $(V_{[\![\ell'-1,\ell]\!]}, x)$, where $V_{[\![\ell'-1,\ell]\!]} \in \mathbb{V}_{[\![\ell'-1,\ell]\!]}$ are the weights, and $x \in \mathcal{H}$ are the inputs, we distinguish the gradients with respect to each of these variables. We denote the gradient with respect to the weights by $\nabla F^{\ell',\ell}$ and we denote the gradient with respect to the inputs by $\mathtt{D}F^{\ell',\ell}$.

Second: In computing the gradient, we want to take advantage of the "compositional nature" of the transition function. To express this in the notation we define a gradient ∇^ℓ to denote the gradient with respect to the variable v^ℓ which lives in the ℓ-th component of the decomposition $\mathbb{V} = \underbrace{\mathcal{V} \oplus \cdots \oplus \mathcal{V}}_{L \text{ times}}$.

With all of this out of the way, we compute step by step the derivatives: First we have:

$$\begin{aligned} [\nabla^\ell S(v)]_m &= \frac{\partial}{\partial [v^\ell]_m} S(v)\\ &= \sum_{f^{\text{in}}\in\mathcal{H}^{\text{in}}} \frac{\partial}{\partial [v^\ell]_m} \sum_n (F_n(V, f^{\text{in}}) - [F_\infty]_n)^2\\ &= \sum_{f^{\text{in}}\in\mathcal{H}^{\text{in}}} \sum_n 2(F_n(V, f^{\text{in}}) - [F_\infty]_n) \frac{\partial}{\partial [v^\ell]_m} F_n(V, f^{\text{in}})\\ &= \sum_{f^{\text{in}}\in\mathcal{H}^{\text{in}}} \sum_n \Big(\frac{\partial}{\partial [v^\ell]_m} F_n(V, f^{\text{in}})\Big) 2(F_n(V, f^{\text{in}}) - [F_\infty]_n)\\ &= \sum_{f^{\text{in}}\in\mathcal{H}^{\text{in}}} \sum_n 2[(\nabla^\ell F)(V, f^{\text{in}})^{\mathtt{t}}]_{mn} (F_n(V, f^{\text{in}}) - [F_\infty]_n), \end{aligned}$$

where the superscript $\mathtt{t}$ denotes the matrix-transposition. We have therefore:

$$\nabla^\ell S(V) = \sum_{f^{\text{in}}\in\mathcal{H}^{\text{in}}} 2(\nabla^\ell F)(V)^{\mathtt{t}} (F(V, f^{\text{in}}) - F_\infty), \quad V \in \mathbb{V}, \quad \ell \in \{0,\dots,L-1\}, \tag{10}$$

where on the right-hand side we have the standard "row-column product" of the matrix $2(\nabla^\ell F)(V)^{\mathtt{t}}$ with the (column) vector $(F(V, f^{\text{in}}) - F_\infty)$.

Second, we want to compute $\nabla^\ell F$. Because of the composition property (5) of the filtered transition functions, we can apply the chain rule (9). The straight forward specialization of that formula to our case reeds[1]:

$$\begin{aligned} \nabla^{\ell-1} F(V, f^{\text{in}}) &= \nabla^{\ell-1} F^{L,\ell}([V]_{L-1,\ell}, F^{\ell,0}([V]_{\ell-1,0}, f^{\text{in}})) \\ &= (\mathtt{D}F^{L,\ell})([V]_{L-1,\ell}, x^\ell)(\nabla^{\ell-1} F^{\ell,0})([V]_{\ell-1,0}, f^{\text{in}}), \end{aligned} \tag{12}$$

where, as before, $x^\ell we. = F^{\ell,0}([V]_{\ell-1,0}, f^{\text{in}})$.

In formula (12) the term $(\nabla^{\ell-1} F^{\ell,0})([V]_{\ell-1,0}, f^{\text{in}})$ can already be written down explicitly. To write it down we recall that the weight v^ℓ is composed of two parameters: $v^\ell = (b^\ell, w^\ell)$. Hence we denote by ∇_b^ℓ, respectively ∇_w^ℓ, the gradient with respect to b^ℓ, respectively w^ℓ. We obtain, for $\ell \in \{1, \dots, L\}$,

$$\begin{aligned} (\nabla_b^{\ell-1} F^{\ell,0})([V]_{\ell-1,0}, f^{\text{in}}) &= \mathbb{1}_{\mathcal{H}\to\mathcal{H}}, \\ (\nabla_w^{\ell-1} F^{\ell,0})([V]_{\ell-1,0}, f^{\text{in}}) &= \Phi^{\ell-1}(F^{\ell-1,0}([V]_{\ell-2,0}, f^{\text{in}})^{\mathcal{H}\leftarrow\mathcal{W}}, \end{aligned} \tag{13}$$

where $\mathbb{1}_{\mathcal{H}\to\mathcal{H}}$ denotes the identity $N\times N$ matrix, that is the identity on $\mathcal{H}$, and $\Phi^{\ell-1}(F^{\ell-1,0}([V]_{\ell-2,0}, f^{\text{in}})^{\mathcal{H}\leftarrow\mathcal{W}}$ denotes the map $\mathcal{W} \to \mathcal{H}$ given in components by

$$\Phi^{\ell-1}(F^{\ell-1,0}([V]_{\ell-2,0}, f^{\text{in}})^{\mathcal{H}\leftarrow\mathcal{W}}(u) = u\, \Phi^{\ell-1}(F^{\ell-1,0}([V]_{\ell-2,0}, f^{\text{in}}),$$

where on the right-hand side we consider the matrix product of the matrix $u \in \mathcal{W}$, thought of as a linear map $u : \mathcal{H} \to \mathcal{H}$, with the vector $\Phi^{\ell-1}(F^{\ell-1,0}([V]_{\ell-2,0}, f^{\text{in}}) \in \mathcal{H}$. In components we have

$$[\Phi^{\ell-1}(F^{\ell-1,0}([V]_{\ell-2,0}, f^{\text{in}})^{\mathcal{H}\leftarrow\mathcal{W}}]_{m,n_1,n_2} = \delta_{mn_1}[\Phi^{\ell-1}(F^{\ell-1,0}([V]_{\ell-2,0}, f^{\text{in}})]_{n_2}, \quad m, n_1, n_2 \in \{1, \dots, N\},$$

where m is the index of a one dimensional array representing a vector $x \in \mathcal{H}$ whereas n_1, n_2 are indices of a two dimensional array representing an element $w \in \mathcal{W}$. At the moment the notation in (13) is the best I could come up with, but it doesn't look completely satisfactory. One should maybe first start by introducing a pairing between elements in $\mathcal{W}$ and elements in $\mathcal{H}$ (the standard product of a matrix with a column vector) and then talk about the "dual" with respect to such a pairing, but I'm still unsure.

Third: The term $(DF^{L,\ell})[V_{[\![L-1,\ell]\!]}, x^\ell)$, on the right-hand side of (12), can be computed by applying again the chain rule (9). In this way we get the following recursive relation which is the origin of the name *back-propagation*:

$$(DF^{L,\ell-1})(V_{[\![L-1,\ell-1]\!]}, x^{\ell-1}) = (DF^{L,\ell})(V_{[\![L-1-1,\ell]\!]}, x^\ell)(DX^\ell_{v^{\ell-1}})(x^{\ell-1}), \tag{14}$$

where we have used the fact that

$$x^\ell = X^\ell_{v^{\ell-1}}(x^{\ell-1}),$$

and where, as before, on the right-hand side of (14), we have the denoted matrix product of the two Jacobian matrices simply by juxtaposition. Note that the Jacobian matrix $DX^{\ell-1}_{v^{\ell-2}}(x^{\ell-2})$ in (14) can be computed explicitly. We get, by linearity of the gradient,

$$\begin{aligned} [(DX^{\ell-1}_{v^{\ell-2}})(x^{\ell-2}))]_{mn} &= \frac{\partial}{\partial x_n}\big(b_m^{\ell-2} + \sum_k w_{mk}^{\ell-2}\Phi_k^{\ell-1}(x)\big)\Big|_{x=x^{\ell-2}} \\ &= \sum_k w_{mk}^{\ell-2}\frac{\partial}{\partial x_n}\Phi_k^{\ell-1}(x)\Big|_{x=x^{\ell-2}} \\ &= \sum_k w_{mk}^{\ell-2}[D\Phi^{\ell-1}(x^{\ell-2})]_{kn} \\ &= [w^{\ell-2}\,(D\Phi^{\ell-1})(x^{\ell-2})]_{mn}. \end{aligned}$$

[1] This is a complicated way to write:

$$\begin{aligned} \frac{\partial F_k(v)}{\partial v^\ell} &= \frac{\partial x^L}{\partial v^\ell} \\ &= \frac{\partial x^L}{\partial x^{\ell+1}}\frac{\partial x^{\ell+1}}{\partial v^\ell}. \end{aligned}$$

In this way we reduce the problem of computing the gradient ∇F to the problem of computing the partial derivatives $\partial x^L/\partial x^{\ell+1}$. These derivatives can be computed recursively (applying again the chain rule). This is the origin of the name *back-propagation*. Indeed, we have

$$\frac{\partial x^{L+1}}{\partial x^{\ell+1}} = \frac{\partial x^{L+1}}{\partial x^{\ell+2}}\frac{\partial x^{\ell+1}}{\partial x^{\ell+1}}. \tag{11}$$

We rewrite this in terms of the more rigorous notation employing the functions $F^{\ell',\ell}$.

Note that

$$
\begin{aligned}
[D\Phi^\ell(x^{\ell-1})]_{kn} &= \frac{\partial}{\partial x_n}\varphi^\ell(x_k)\Big|_{x=x^{\ell-1}} \\
&= (\varphi^\ell)'(x_k)\frac{\partial x_k}{\partial x_n}\Big|_{x=x^{\ell-1}} \\
&= (\varphi^\ell)'(x_k)\delta_{nk}\Big|_{x=x^{\ell-1}} \\
&= (\varphi^\ell)'([x^{\ell-1}]_n)\delta_{nk} \quad \text{(no sum over repeated indices)},
\end{aligned} \tag{15}
$$

where the prime ($'$) denotes the derivative of the function. Hence, (14) becomes:

$$
(DF^{L,\ell-1})(V_{[\![L-1,\ell-1]\!]}, x^{\ell-1}) = (DF^{L,\ell})(V_{[\![L-1,\ell]\!]}, x^\ell)\, w^{\ell-1}\,(D\Phi^\ell)(x^{\ell-1}). \tag{16}
$$

To obtain this formula we had to express each object as a function, so that we could differentiate with respect to its argument. Now, to clarify the implementation of this formula, we go back to a "declarative" notation as in (11). We fix the weights $V \in \mathbb{V}$ and the inputs $f^{\text{in}} \in \mathcal{H}^{\text{in}}$, then we let

$$
\mathbf{J}^\ell := (DF^{L,\ell})(V_{[\![L-1,\ell]\!]}, x^\ell)), \quad \ell \in \{0,\dots,L-1\}, \tag{17}
$$

where as usual the x^ℓ are the constant vectors of (2) and (6). Hence, from formula (16), we get the "back-propagating" recursive relation

$$
\mathbf{J}^{\ell-1} = \mathbf{J}^\ell\, w^{\ell-1}\,(\mathtt{D}\Phi^\ell)(x^{\ell-1}), \quad \ell \in \{1,\dots,L+1\}. \tag{18}
$$

This formula is "backward-propagating" in the sense that to compute the gradient of the function $F^{\text{out},\ell+1}$ which takes as input the outputs of the layer ℓ we use the gradient of the function $F^{\text{out},\ell+2}$ which takes as inputs the outputs of the following layer $\ell+1$, this means that we are "propagating backward" from the layer $\ell+1$ to the layer ℓ.

We put everything together in a formula ready for implementation. We revert to the common convention where the weights for a given layer are thought of as describing a "state" of that layer instead of being thought of as an additional set of inputs for such a layer.

Proposition 1.2. *Fix $M, L \in \mathbb{N}$. Let, as in section 1.3, the input space $\mathcal{H}$ and the weight space $\mathbb{V}$ be defined as follows:*

$$
\begin{aligned}
&\mathcal{H} \stackrel{\text{def}}{=} \mathbb{R}^N, \\
&\mathbb{V} \stackrel{\text{def}}{=} \mathbb{B} \oplus \mathbb{W} = \oplus_{\ell=0}^L \mathcal{V}, \quad \mathbb{B} \stackrel{\text{def}}{=} \oplus_{\ell=0}^L \mathcal{B} \cong \mathbb{R}^N \times \mathbb{R}^L, \quad \mathbb{W} \stackrel{\text{def}}{=} \oplus_{\ell=0}^L \mathcal{W} \cong \mathbb{R}^N \times \mathbb{R}^N \times \mathbb{R}^L \\
&\mathcal{V} \stackrel{\text{def}}{=} \mathcal{B} \oplus \mathcal{W}, \quad \mathcal{B} \stackrel{\text{def}}{=} \mathbb{R}^N, \quad \mathcal{W} \stackrel{\text{def}}{=} \mathbb{R}^N \times \mathbb{R}^N.
\end{aligned}
$$

For $\ell \in \{1,\dots,L\}$, we denote by $\varphi^\ell : \mathbb{R} \to \mathbb{R}$ be the activation function of a neuron on the ℓ-th layer and we define the functions $\Phi^\ell : \mathcal{H} \to \mathcal{H}$ by

$$
[\Phi^\ell(x)]_n = \varphi([x]_n), \quad x \in \mathcal{H},
$$

where $[y]_n$ denotes the n-th component of a vector $y \in \mathcal{H}$. Let the characteristic function $F : \mathbb{V} \times \mathcal{H} \to \mathcal{H}$ be defined as follows. Fix a given input $f \in \mathcal{H}$ and a collection of weights $V \in \mathbb{V}$, $V = (v^0,\dots,v^L)$, where, for $\ell \in \{0,\dots,L\}$, $v^\ell = (b^\ell, w^\ell)$, $b^\ell \in \mathcal{B}$, $w^\ell \in \mathcal{W}$. Then, we define recursively a collection of "partial inputs" $(x^\ell)_{\ell\in\{0,\dots,L+1\}}$, $x^\ell \in \mathcal{H}$:

$$
\begin{aligned}
&x^\ell \stackrel{\text{def}}{=} b^\ell + w^\ell \Phi^\ell(x^{\ell-1}), \quad \ell \in \{1,\dots,L\}, \\
&x^0 = f.
\end{aligned} \tag{19}
$$

Finally, we set (cf. (11)) $F(V,f) = x^L$. Let the cost function $S : \mathbb{V} \to \mathbb{R}$ be:

$$
S(V) \stackrel{\text{def}}{=} \sum_{f\in\mathcal{H}} \|F(f,V) - F_\infty(f)\|^2.
$$

Then the collection of gradients $\nabla S(V)$ is obtained via the following relations, with $\ell \in \{1,\dots,L\}$,

$$
\begin{aligned}
[\mathtt{D}\Phi^\ell(x^{\ell-1})]_{mn} &= \varphi'([a]_n)\delta_{mn} \quad \textit{(no sum over repeated indices)}\ , \\
\mathbf{J}^{\ell-1} &= \prod_{\ell'=L}^{\ell} \left(w^\ell\, (\mathtt{D}\Phi^{\ell'})(x^{\ell'-1}) \right) = \mathbf{J}^\ell\, w^\ell\, (\mathtt{D}\Phi^\ell)(x^{\ell-1}), \quad \mathbf{J}^L = \mathbb{1}, \\
\nabla_b^\ell F(V,f) &= \mathbf{J}^\ell, \\
\nabla_w^\ell F(V,f) &= \mathbf{J}^\ell \Phi^\ell(x^{\ell-1}), \\
\nabla S(V) &= 2(\nabla F)(V,f)^{\mathtt{t}}\, (F(V,f) - F_\infty(f)).
\end{aligned} \tag{20}
$$

Proof. We point to where the relations in (20) were obtained by "back-propagating" in this section (:^). The first line was obtained in (15); the second line is consequence of (16), (17), (18); the third and fourth lines are a consequence of (12) and (13); the forth line was obtained in (10) . □

Corollary 1.2.1. *With the same notation as the theorem, we have the following.*

$$
\begin{aligned}
\nabla_b^{\ell-1} S(V) &= (\mathtt{D}\Phi^\ell)(x^{\ell-1})^{\mathtt{t}}(w^\ell)^{\mathtt{t}}\nabla_b^\ell S(V), \\
\nabla_w^{\ell-1} S(V) &= 2(\Phi^\ell(x^{\ell-1})^{\mathcal{H}\leftarrow\mathcal{W}})^{\mathtt{t}}\, (\nabla_b^\ell S)(V),
\end{aligned}
$$

where the transposition in the last line in components reads

$$
[(\Phi^\ell(x^{\ell-1})^{\mathcal{H}\leftarrow\mathcal{W}})^{\mathtt{t}}]_{n_1,n_2,m} = [\Phi^\ell(x^{\ell-1})^{\mathcal{H}\leftarrow\mathcal{W}}]_{m,n_1,n_2}, \quad m,n_1,n_2 \in \{1,\dots,N\}.
$$

Proof. We have

$$
\begin{aligned}
\nabla_b^{\ell-1} S(V) &= 2(\mathbf{J}^{\ell-1})^{\mathtt{t}}(F(V,f) - F_\infty(f)) \\
&= (\mathtt{D}\Phi^\ell)(x^{\ell-1})^{\mathtt{t}}(w^\ell)^{\mathtt{t}}\mathbf{J}^\ell(F(V,f) - F_\infty(f)) \\
&= (\mathtt{D}\Phi^\ell)(x^{\ell-1})^{\mathtt{t}}(w^\ell)^{\mathtt{t}}\nabla_b^\ell S(V), \\
\nabla_w^{\ell-1} S(V) &= 2(\Phi^\ell(x^{\ell-1})^{\mathcal{H}\leftarrow\mathcal{W}})^{\mathtt{t}}\Phi^\ell(x^{\ell-1})^{\mathtt{t}}(\mathbf{J}^{\ell-1})^{\mathtt{t}}(F(V,f) - F_\infty(f)) \\
&= 2(\Phi^\ell(x^{\ell-1})^{\mathcal{H}\leftarrow\mathcal{W}})^{\mathtt{t}}(\mathtt{D}\Phi^\ell)(x^{\ell-1})^{\mathtt{t}}(w^\ell)^{\mathtt{t}}\nabla_b^\ell S(V).
\end{aligned}
$$

□

References

[Boo03] William M. Boothby, *An introduction to differentiable manifolds and Riemannian geometry*, revised second ed., vol. 120, Academic Press, 2003.

[GBC16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville, *Deep learning*, MIT press, 2016.