Machine Perception - Pen & Paper Solutions
Backpropagation

1 Multivariable chain rule

The gradient $\nabla_{\boldsymbol{x}}\mathbf{f}$ of a function $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}$, $\mathbf{f} : \boldsymbol{x} \mapsto y$ is defined as $\nabla_{\boldsymbol{x}}\mathbf{f} = [\frac{\partial y}{\partial x_1}, \frac{\partial y}{\partial x_2}, \ldots, \frac{\partial y}{\partial x_n}]^\intercal \in \mathbb{R}^n$. The Jacobian $J(g)_{\boldsymbol{x}}$ of a function $\mathbf{g} : \mathbb{R}^n \to \mathbb{R}^m$, $\mathbf{g} : \boldsymbol{x} \mapsto \boldsymbol{y}$ is defined as a matrix of size $\mathbb{R}^{m \times n}$, where each element $J(g)_{\boldsymbol{x},ij}$ is defined as $J(g)_{\boldsymbol{x},ij} = \frac{\partial y_i}{\partial x_j}$. Therefore, the *transposed* gradient can be regarded as a special case of the Jacobian, where $m = 1$.
The chain-rule in one dimension, for functions $f, g : \mathbb{R} \to \mathbb{R}$, $g(x) = y$, $f(y) = z$, is expressed as the following:

$$\frac{\partial f(g(x))}{\partial x} = \frac{\partial f(g(x))}{\partial g(x)} \frac{\partial g(x)}{\partial x} = \frac{\partial z}{\partial y} \frac{\partial y}{\partial x} \tag{1}$$

In higher dimensions, where $\mathbf{g} : \mathbb{R}^n \to \mathbb{R}^m$, $\boldsymbol{x} \mapsto \boldsymbol{y}$ and $\mathbf{f} : \mathbb{R}^m \mapsto \mathbb{R}^l$, $\boldsymbol{y} \mapsto \boldsymbol{z}$, the chain rule states the following:

$$\frac{\partial z_k}{\partial x_j} = \sum_{i=1}^{m} \frac{\partial z_k}{\partial y_i} \frac{\partial y_i}{\partial x_j} \tag{2}$$

1.1 Chain rule for Jacobians

Given the definition above, show that $\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{x}} = \frac{\partial \boldsymbol{z}}{\partial \boldsymbol{y}} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}$. What are the dimensions?
Hint: First derive $\frac{\partial z_k}{\partial \boldsymbol{x}}$

We have $\frac{\partial z_k}{\partial \boldsymbol{x}} = [\frac{\partial z_k}{\partial x_1}, \frac{\partial z_k}{\partial x_2}, \ldots, \frac{\partial z_k}{\partial x_n}] \in \mathbb{R}^n$ and $\frac{\partial z_k}{\partial x_j} = \sum_{i=1}^{m} \frac{\partial z_k}{\partial y_i} \frac{\partial y_i}{\partial x_j} = \frac{\partial z_k}{\partial \boldsymbol{y}} \frac{\partial \boldsymbol{y}}{\partial x_j}$, hence:

$$\frac{\partial z_k}{\partial \boldsymbol{x}} = [\frac{\partial z_k}{\partial x_1}, \frac{\partial z_k}{\partial x_2}, \ldots, \frac{\partial z_k}{\partial x_n}] = [\frac{\partial z_k}{\partial \boldsymbol{y}} \frac{\partial \boldsymbol{y}}{\partial x_1}, \frac{\partial z_k}{\partial \boldsymbol{y}} \frac{\partial \boldsymbol{y}}{\partial x_2}, \ldots, \frac{\partial z_k}{\partial \boldsymbol{y}} \frac{\partial \boldsymbol{y}}{\partial x_n}] = [\frac{\partial z_k}{\partial \boldsymbol{y}} \left(\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\right)_{:,1}, \frac{\partial z_k}{\partial \boldsymbol{y}} \left(\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\right)_{:,2}, \ldots, \frac{\partial z_k}{\partial \boldsymbol{y}} \left(\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\right)_{:,n}] = \frac{\partial z_k}{\partial \boldsymbol{y}} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}$$

Similarly, we have for $\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{x}}$ using the above results:

$$\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{x}} = \begin{bmatrix} \frac{\partial z_1}{\partial \boldsymbol{y}} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}} \\ \frac{\partial z_2}{\partial \boldsymbol{y}} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}} \\ \vdots \\ \frac{\partial z_l}{\partial \boldsymbol{y}} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}} \end{bmatrix} = \frac{\partial \boldsymbol{z}}{\partial \boldsymbol{y}} \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}} \in \mathbb{R}^{l \times n}$$

In other words, the chain rule has the same form for higher dimensional functions as in the one-dimensional case.

1.2 Chain rule for graphs

We saw the multivariable chain rule defined for higher dimensional functions. In term of graphs, f, g can be visualized as follows:

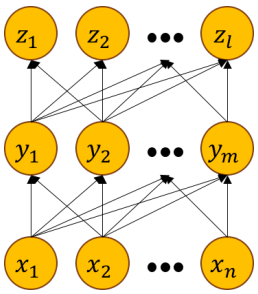
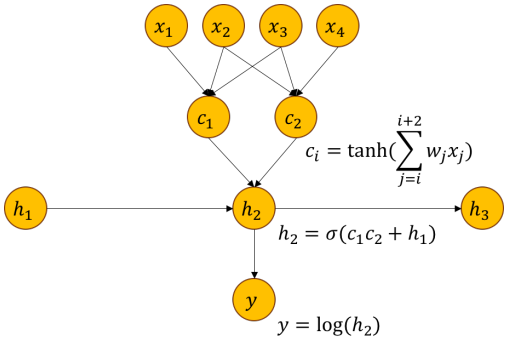


Figure 1: Visualization of the functions f and g as graphs.

Therefore, for a given z_k and x_i, one can imagine the multivariable chain rule as the sum of univariable chain rules over all possible child nodes of x_i that connects it to z_k. Neural networks can be regarded as graphs, where a node outputs values that are a function of its inputs. The input is defined by the edges that connect nodes to each other. Using this, it is possible to define complicated architectures that are still fully differentiable, hence can be optimized.
Let $\sigma = \frac{1}{1+\exp(-x)}$ be the sigmoid activation and $\tanh = \frac{\exp(x)-\exp(-x)}{\exp(x)+\exp(-x)}$. Given the following graph:



a) Derive $\frac{\partial y}{\partial x_2}$

We define $(\boldsymbol{X})_{:,i}$ as selecting the i-th column of the matrix $\boldsymbol{X}$. Using the chain rule, we get:

$$\frac{\partial y}{\partial x_2} = \frac{\partial y}{\partial h_2}\left(\frac{\partial h_2}{\partial c_1}\frac{\partial c_1}{\partial x_2} + \frac{\partial h_2}{\partial c_2}\frac{\partial c_2}{\partial x_2}\right)$$

Where each partial derivative has the following values:

$$\frac{\partial}{\partial x}\tanh(x) = 1 - \tanh^2(x), \quad \frac{\partial}{\partial x}\sigma(x) = \sigma(x)(1-\sigma(x)), \quad \frac{\partial}{\partial x}\log(x) = \frac{1}{x}$$

$$\frac{\partial y}{\partial h_2} = \frac{1}{h_2}, \quad \frac{\partial h_2}{\partial c_1} = h_2(1-h_2)c_2, \quad \frac{\partial h_2}{\partial c_2} = h_2(1-h_2)c_1, \quad \frac{\partial c_1}{\partial x_2} = (1-c_1^2)w_2, \quad \frac{\partial c_2}{\partial x_2} = (1-c_2^2)w_2$$

Putting the partial derivatives together, we get:

$$\frac{\partial y}{\partial x_2} = \frac{1}{h_2}\Big(h_2(1-h_2)c_2(1-c_1^2)w_2 + h_2(1-h_2)c_1(1-c_2^2)w_2\Big) = (1-h_2)w_2\Big(c_1 + c_2 - c_2c_1^2 - c_1c_2^2\Big)$$

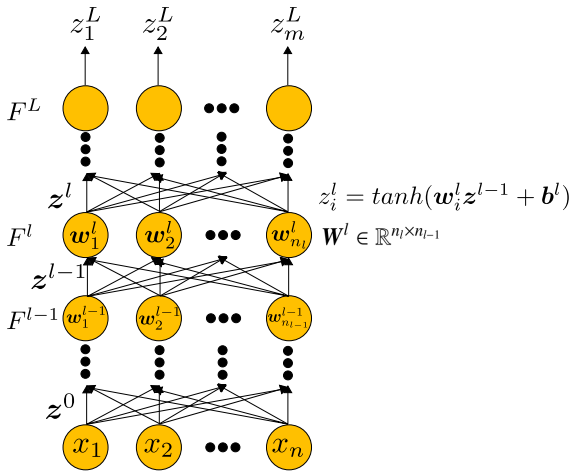


Figure 2: Visulization of multiplayer perceptron with L layers.

2 Backpropagation in multilayer perceptron

A multilayer perceptron, having L layers, can be defined as:

$$F = F^L \circ F^{L-1} \circ \cdots \circ F^1 : \mathbb{R}^n \to \mathbb{R}^m, F(\boldsymbol{x}) = \boldsymbol{y} = \boldsymbol{z}^L \tag{3}$$

Where $\circ$ is the composition operator (i.e $f(g(x)) = f \circ g(x)$), $F^l = \tanh \circ \tilde{F}^l$, where $\tilde{F}^l(\boldsymbol{x}) = \boldsymbol{W}^l\boldsymbol{x} + \boldsymbol{b}^l$ with $\boldsymbol{W}^l \in \mathbb{R}^{n_l \times n_{l-1}}$, $\boldsymbol{b}^l \in \mathbb{R}^{n_l}$, and $\tanh(\boldsymbol{x}) = \frac{\exp(x)-\exp(-x)}{\exp(x)+\exp(-x)}$ is the tanh activation function applied element-wise and $\boldsymbol{z}^0 = \boldsymbol{x}$. Let $\mathcal{R} : \mathbb{R}^m \to \mathbb{R}$ be a function on $\boldsymbol{z}^L$ computing the loss. We define the error signal of layer l as $\boldsymbol{\delta}^l := \frac{\partial \mathcal{R}}{\partial \boldsymbol{z}^l}$, where $\boldsymbol{z}^l = F^l(\boldsymbol{z}^{l-1})$. We left out $\boldsymbol{z}_L$ from $\mathcal{R}$ to reduce clutter, but still assume its there.

a) For a given layer l, derive $\boldsymbol{\delta}^l$ as a function of error signals $\boldsymbol{\delta}^k$ and Jacobians $\frac{\partial \boldsymbol{z}^k}{\partial \boldsymbol{z}^{k-1}}$

We have $\boldsymbol{\delta}^l = \frac{\partial \mathcal{R}}{\partial \boldsymbol{z}^l}$ per definition. Expanding this yields:

$$\boldsymbol{\delta}^l = \frac{\partial \mathcal{R}}{\partial \boldsymbol{z}^l} = \frac{\partial}{\partial \boldsymbol{z}^l}\Big(\mathcal{R} \circ F^L \circ ... \circ F^{l+1}(z^l)\Big) = \boldsymbol{\delta}^L \cdot \frac{\partial \boldsymbol{z}^L}{\partial \boldsymbol{z}^{L-1}} \cdot \ldots \cdot \frac{\partial \boldsymbol{z}^{l+1}}{\partial \boldsymbol{z}^l} = \boldsymbol{\delta}^{l+1} \cdot \frac{\partial \boldsymbol{z}^{l+1}}{\partial \boldsymbol{z}^l}$$

b) Derive $\frac{\partial \mathcal{R}}{\partial w_{ij}^l}$ and $\frac{\partial \mathcal{R}}{\partial b_i^l}$ as a function of $\boldsymbol{\delta}^l$, $\boldsymbol{W}^l$, $\boldsymbol{b}^l$ and $\boldsymbol{z}^l$. Which term was backpropagated from layer $l+1$ to layer l?

Using the definition of $\mathcal{R}$, we get:

$$\frac{\partial \mathcal{R}}{\partial w_{ij}^l} = \frac{\partial}{\partial w_{ij}^l}\Big(\mathcal{R} \circ F^L \circ ... \circ F^{l+1}(\boldsymbol{z}^l)\Big) = \frac{\partial}{\partial w_{ij}^l}\Big(\mathcal{R} \circ F^L \circ ... \circ F^{l+1}(\tanh(\boldsymbol{W}^l\boldsymbol{x}^{l-1} + \boldsymbol{b}^l))\Big)$$

Applying the chain rule, we get:

$$\frac{\partial}{\partial w_{ij}^l}\Big(\mathcal{R} \circ F^L \circ ... \circ F^{l+1}(\boldsymbol{z}^l)\Big) = \frac{\partial \mathcal{R}}{\partial \boldsymbol{z}^l}\frac{\partial \boldsymbol{z}^l}{\partial w_{ij}^l} = \boldsymbol{\delta}^l \frac{\partial \boldsymbol{z}^l}{\partial w_{ij}^l}$$

Where $\frac{\partial \mathcal{R}}{\partial \boldsymbol{z}^l} = \boldsymbol{\delta}^l$ as derived in **a)**. To determine $\frac{\partial \boldsymbol{z}^l}{\partial w_{ij}^l}$ let's define $\boldsymbol{z}^l = \tanh(\boldsymbol{W}^l\boldsymbol{z}^{l-1} + \boldsymbol{b}^l) =: \tanh(\boldsymbol{a})$. Recall that $\boldsymbol{z}^l$ is a column vector, hence the derivative w.r.t. a scalar must also be a column vector. Using the chain rule we can write.

$$\frac{\partial \boldsymbol{z}^l}{\partial w_{ij}^l} = \frac{\partial \tanh(\boldsymbol{a})}{\partial \boldsymbol{a}}\frac{\partial \boldsymbol{a}}{\partial w_{ij}^l}$$

The first factor is the Jacobian, which in this case is a diagonal matrix, because when we derive a_j by a_i it will be zero unless $i = j$. Lets denote $\tanh'(x) = 1 - \tanh^2(x)$ and recall that tanh and its derivative just operate element-wise for vector-valued input. It follows that

$$\left[\frac{\partial \tanh(\boldsymbol{a})}{\partial \boldsymbol{a}}\right]_{ij} = \frac{\partial a_i}{\partial a_j} = \tanh'(a_i) \quad \text{if } i = j,\ 0 \text{ otherwise}$$

and thus

$$\frac{\partial \tanh(\boldsymbol{a})}{\partial \boldsymbol{a}} = \begin{bmatrix} \tanh'(a_1) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \tanh'(a_{n_l}) \end{bmatrix} = \text{diag}(1 - \tanh^2(\boldsymbol{a})) \tag{4}$$

To determine $\frac{\partial \boldsymbol{a}}{\partial w^l_{ij}}$ let's expand $\boldsymbol{a}$ again. Remember from linear algebra that a matrix-vector multiplication is nothing more than a linear combination of the matrix's column vectors with the vector's elements as weights:

$$\boldsymbol{a} = \boldsymbol{W}^l \boldsymbol{z}^{l-1} + \boldsymbol{b}^l = z^{l-1}_1 \boldsymbol{w}^l_1 + \ldots + z^{l-1}_j \boldsymbol{w}^l_j + \ldots + z^{l-1}_{n_{l-1}} \boldsymbol{w}^l_{n_{l-1}} + \boldsymbol{b}^l$$

Here, $\boldsymbol{w}^l_j$ refers to the j-th column of $\boldsymbol{W}^l$. If we derive this by w^l_{ij} it is clear that the terms where $\boldsymbol{w}^l_j$ does not occur just equate to 0. Hence we are left with the term $z^{l-1}_j \boldsymbol{w}^l_j$ which is a vector, and, again, taking the element-wise derivative of those elements that do not contain w^l_{ij} (which are all entries except the i-th row) will result in 0. We write this down as follows:

$$\frac{\partial \boldsymbol{a}}{\partial w^l_{ij}} = \underbrace{[0, \cdots, \overbrace{z^{l-1}_j}^{i\text{-th entry}}, \cdots 0]^T}_{n_l \text{ many entries}} =: \text{vec}^{n_l}_i(z^{l-1}_j) \tag{5}$$

If you now multiply equations 4 and 5 to obtain the desired gradient you will see that lots of terms cancel out due to the sparseness of both factors. We end up with

$$\frac{\partial \boldsymbol{z}^l}{\partial w^l_{ij}} = \text{vec}^{n_l}_i(1 - \tanh^2(a_i)) \cdot z^{l-1}_j$$

which we can also write as

$$\left[\frac{\partial \boldsymbol{z}^l}{\partial w^l_{ij}}\right]_k = (1 - \tanh^2(\boldsymbol{a}))_k \cdot z^{l-1}_j \quad \text{if } k = i,\ 0 \text{ otherwise}$$

Similarly, we get for $\frac{\partial \mathcal{R}}{\partial b^l_i}$:

$$\frac{\partial \mathcal{R}}{\partial b^l_i} = \frac{\partial \mathcal{R}}{\partial \boldsymbol{z}^l} \frac{\partial \boldsymbol{z}^l}{\partial b^l_i} = \boldsymbol{\delta}^l \frac{\partial \boldsymbol{z}^l}{\partial b^l_i}$$

with (differentiating this is very similar to the above derivations):

$$\frac{\partial \boldsymbol{z}^l}{\partial b^l_i} = \text{vec}^{n_l}_i(1 - \tanh^2(a_i))$$