DERIVATION OF BACKPROPAGATION

1. Notation

Suppose we have a neural network built for a binary classification task (see Fig. 1 for example). For binary classification we are solving an optimization task:

$$F(b) = -\frac{1}{N}\sum_{k=1}^{N}\big[y_k \ln p_k + (1-y_k)\ln(1-p_k)\big] \to \min_b,$$

where N — batch size, y_k — true label of the k-th object, $p_k = a_L(x_k)$ — output of the network on the k-th object, b — set of neural network parameters.

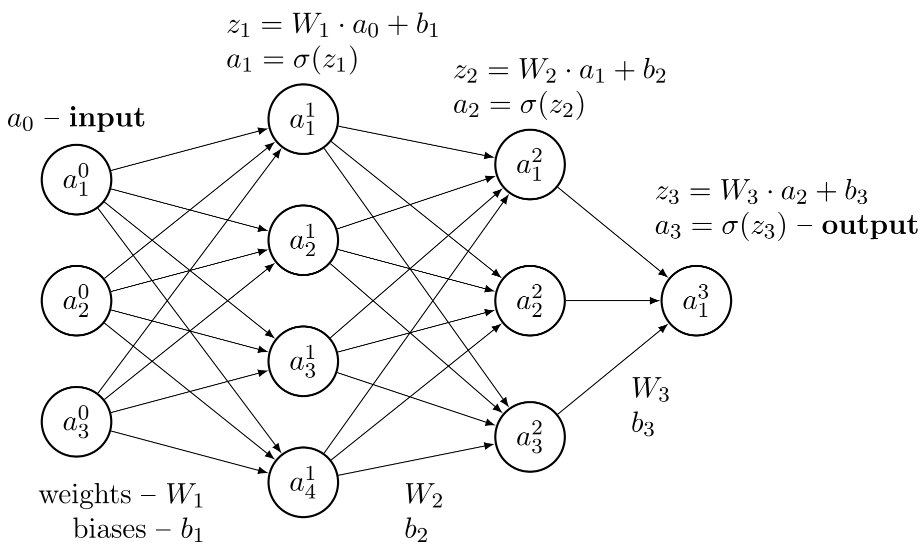

Figure 1. A feed-forward neural network architecture

We are going to use the following notation:

- 0 — input layer, L — output layer,
- n_l — l-th layer size,
- $a_l = (a_i^l)_{(n_l\times 1)}$ — l-th layer output, a_0 — input,
- $W_l = (w_{ij}^l)_{(n_i\times n_{i-1})}$ — weights, $b_l = (b_i^l)_{(n_l\times 1)}$ — biasses connecting l-th layer with $(l-1)$-th layer,
- $z_l = W_l \cdot a_{l-1} + b_l$ — l-th layer before activation,
- $\sigma(x) = \dfrac{1}{1+e^{-x}}$ — sigmoid function, $\sigma'(x) = \sigma(x)(1-\sigma(x))$,
- «$*$» — element-size multiplication, «$\cdot$» – matrix multiplication.

2. Gradient derivation

In order to calculate the gradient vector ∇F, we need to calculate each partial derivative $\dfrac{\partial F}{\partial w_{ij}^l}$ and $\dfrac{\partial F}{\partial b_i^l}$.

We start with the derivative

$$\frac{\partial F}{\partial z_L} = \frac{\partial F}{\partial a_L}\frac{\partial a_L}{\partial z_L}$$

Strictly speaking, this is a matrix, nevertheless the non-diagonal derivatives (i.e. $\frac{\partial a_i^L}{\partial z_j^L}$ for $i \neq j$) are zeros, thus, in fact this is a vector of size n_L.

First, suppose the batch size is $N = 1$. In this case

$$F = -y \ln a_L - (1-y)\ln(1-a_L)$$

and

$$\frac{\partial F}{\partial a_L} = -\frac{y}{a_L} + \frac{1-y}{1-a_L} = \frac{-y(1-a_L) + (1-y)a_L}{a_L(1-a_L)} = \frac{a_L - y}{a_L(1-a_L)}$$

For the second factor $\frac{\partial a_L}{\partial z_L}$, since $a_L = \sigma(z_L)$, we have

$$\frac{\partial a_L}{\partial z_L} = \sigma(z_L)(1-\sigma(z_L)) = a_L(1-a_L)$$

Putting it all together, we have

$$\frac{\partial F}{\partial z_L} = a_L - y$$

We are going to get back to this layer later, but for now let us establish how to derive $\frac{\partial L}{\partial z_l}$ using $\frac{\partial L}{\partial z_{l+1}}$ for $l < L$. We have

$$\frac{\partial L}{\partial z_l} = \frac{\partial L}{\partial z_{l+1}} \frac{\partial z_{l+1}}{\partial a_l} \frac{\partial a_l}{\partial z_l},$$

since $z_{l+1} = W_{l+1} \cdot a_l + b_{l+1}$. By induction, we can assume that the first factor $\frac{\partial L}{\partial z_{l+1}}$ has already been calculated on the previous step and is a vector of size n_{l+1}. The second factor $\frac{\partial z_{l+1}}{\partial a_l}$ is a matrix of size $n_{i+1} \times n_l$. In fact, this is exactly the matrix W_{l+1}. Finally, the last factor $\frac{\partial a_l}{\partial z_l}$ is simply the sigmoid derivative. Thus,

$$\frac{\partial F}{\partial z_l} = W_{l+1}^\top \cdot \frac{\partial F}{\partial z_{l+1}} * \big(\sigma(z_l) * (1-\sigma(z_l))\big),$$

where $\sigma(z_l)$ is element-wise.

In this fashion we can calculate each partial derivative of F over z_l. Since each z_l is linear with respect to W_l and b_l, we have:

$$\frac{\partial z_l}{\partial W_l} = a_{l-1}^\top, \ \ \frac{\partial z_l}{\partial b_l} = 1$$

The final formulas:

$$\frac{\partial F}{\partial W_l} = \frac{\partial F}{\partial z_L} \cdot a_{l-1}^\top, \ \ \frac{\partial F}{\partial b_l} = \frac{\partial F}{\partial z_l}$$

Finally, the gradient vector ∇F consists of all the partial derivatives over W_l and b_l, but we do not actually need to put them all into a vector, since we only need their values.