Exercise 2 - Neural Networks

Alexander Bigalke, Arthur Heimbrecht, Robin Rombach

May 7, 2018

I. Linear Activation Function

In a feed-forward network, the output of each layer l (denoted $Z_l \in \mathbb{R}^{m_l}$; with m_l some integer specifying the number of features in layer l.) is computed iteratively by multiplying the weight matrix $B_l \in \mathbb{R}^{m_l \times m_{l-1}}$ with the output of layer $l-1$ and subsequently applying the activation function φ_l to this product:

$$\tilde{Z}_l = B_l \cdot Z_{l-1} \tag{1}$$
$$Z_l = \varphi_l(\tilde{Z}_l) \tag{2}$$

where the input Z_0 is given as

$$Z_0 = \begin{pmatrix} 1 \\ \mathbf{X} \end{pmatrix}$$

with $\mathbf{X}$ denoting the input (training) data.

Let us assume that the activation function is a linear map, i.e.

$$Z_l = \varphi_l(\tilde{Z}_l) = \Phi_l \cdot \tilde{Z}_l \tag{3}$$

such that Φ_l is a quadratic matrix with $\Phi_l \in \mathbb{R}^{m_l \times m_l}$.
Combining equations (1) and (3) and exploiting associativity of matrix multiplication then yields the following output for layer l:

$$Z_l = \Phi_l \cdot \tilde{Z}_l = \Phi_l \cdot B_l \cdot Z_{l-1} \equiv \mathcal{B}_l \cdot Z_{l-1} \ . \tag{4}$$

Hence,

$$\mathbb{R}^{m_l \times m_{l-1}} \ni \mathcal{B}_l = \Phi_l \cdot B_l$$

Expanding (4) gives:

$$Z_l = \underbrace{\mathcal{B}_l \cdot \mathcal{B}_{l-1} \cdot \mathcal{B}_{l-2} \cdot \dots \cdot \mathcal{B}_0}_{\equiv \mathbf{B}_l^0 \in \mathbb{R}^{m_l \times m_0}} \cdot Z_0 \tag{5}$$

Thus, in the case of a linear activation function and by contracting the product $\mathcal{B}_l \cdot \mathcal{B}_{l-1} \cdot \mathcal{B}_{l-2} \cdot \cdots \cdot \mathcal{B}_0$, the whole procedure of mapping the input layer 0 to layer l via several intermediate layers $l - i$ ($i \in [0, l]$) reduces to mapping layer 0 directly to layer l by

$$Z_l = \mathbf{B}_l^0 \cdot Z_0 \tag{6}$$

with $\mathbf{B}_l^0 \in \mathbb{R}^{m_l \times m_0}$.