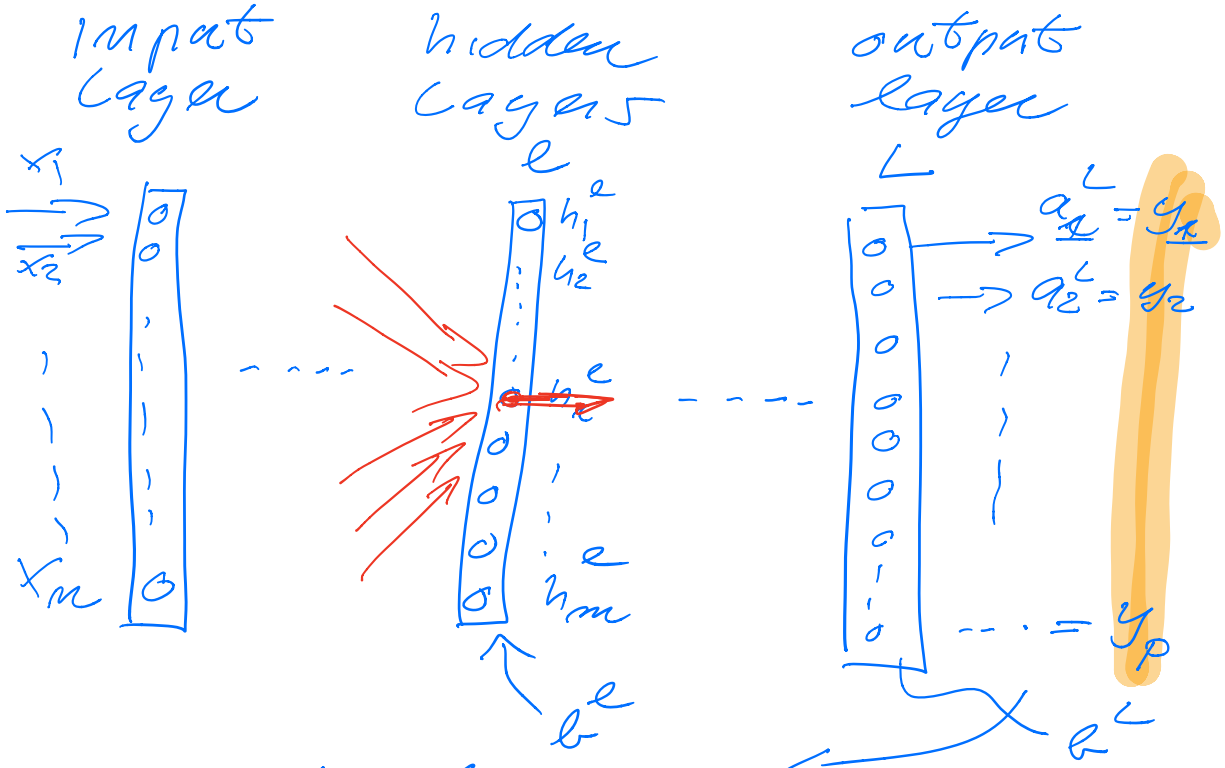
Lecture october 8

FFNN

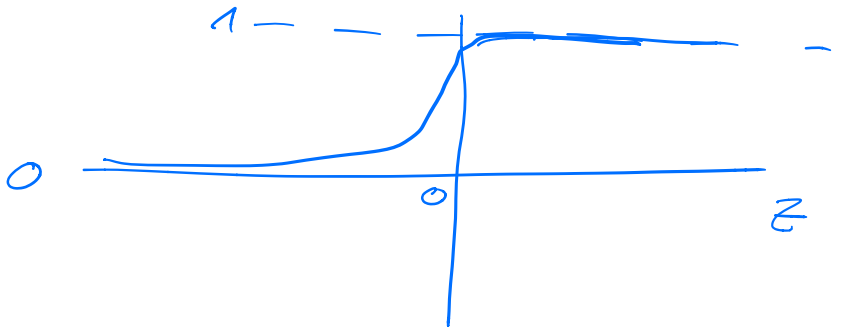


- activation functions

$$a_k^l = f(z_k^l) = f\left(\sum_j w_{kj}^l a_j^{l-1} + b_k^l\right)$$

$$f(z) = \sigma(z) = \frac{e^z}{1+e^z}$$

$$f(z) = \tanh(z) = 2\sigma(2z) - 1$$



Rectified Linear Unit (RELU)

$$f(z) = \max(0, z)$$

output values only for $z > 0$, that are different for zero.

Leaky RELU

$$f(z_i, \alpha_i) = \max(0, z_i) + \alpha_i \min(0, z_i)$$

$\alpha_i \sim 0.01$

ELU

$$f(z) = \begin{cases} \alpha e^{z} - 1 & z < 0 \\ z & z \geq 0 \end{cases}$$

.... many more

output layer with own activation function + Cost.

Cross entropy $\Rightarrow$ Binary classification : $\sigma(z)$
More classes : Softmax

- Regression: MSE as cost function
$\sigma(z)$ as activation.

$$\frac{1}{2}\sum_i (y_i^L - t_i)^2$$

architecture:

- # hidden layers (Depth)
  # nodes in each layer. (width)

Regularization parameters

- Ridge: add to cost function $\frac{\lambda}{2} W^T W$
- Lasso: $\lambda |W|_1$

SGD: epoch, learning, mini-batches

- Standard
- Momentum SGD
- Adagrad
- RMS prop
- Adam
- ⋮

Feed Forward

$$a_i^{(1)} = f^{(1)}\left(W^{(1)} x + b^{(1)}\right)$$

$$a_i^{(2)} = f^{(2)}\left(w^{(2)} a^{(1)} + b^{(2)}\right)$$

$$= f^{(2)}\left(w^{(2)} f^{(1)}\left(w^{(1)} x + b^{(1)}\right) + b^{(2)}\right)$$

$\vdots$

FFNN : the basic operation is $W^{(l)} a^{(l-1)} + b^{(l)}$

Back prop algo

$$\frac{\partial C(w^{(l)})}{\partial w_{jk}^{(l)}} = \delta_j^{(l)} a_k^{(l-1)}$$

$$\delta_j^{(l)} = f'(z_j^{(l)}) \frac{\partial C}{\partial a_j^{(l)}}$$

$$\delta_j^{(l)} = \frac{\partial C}{\partial f_j} e$$

$$\delta_j^{(l)} = \sum_k \delta_k^{(l+1)} w_{kj}^{(l+1)} \cdot f'(z_j^{(l)})$$

Gradient descent

$$w_{jk}^{(l)} \leftarrow w_{jk}^{(l)} - \gamma \delta_j^{(l)} a_k^{(l-1)}$$

$$b_j^{(e)} \leftarrow b_j^{(e)} - \gamma s_j^{(e)}$$