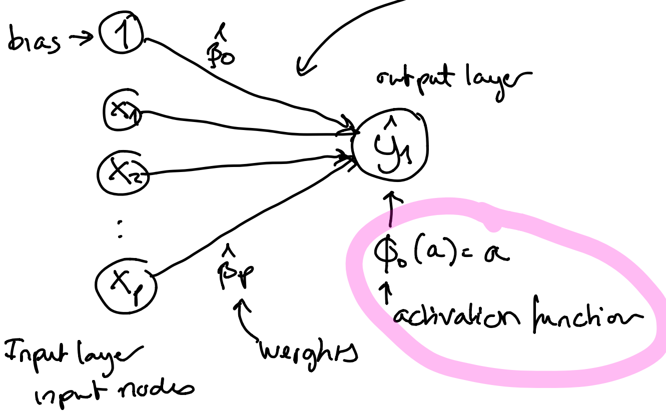
Last time: multiple linear regression $Y_i = \beta_0 + \beta_1 \cdot x_{i1} + \dots + \beta_p x_{ip} + \varepsilon_i$

$$\hat{y}_1(x_i) = \hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \dots + \hat{\beta}_p \cdot x_{ip}$$



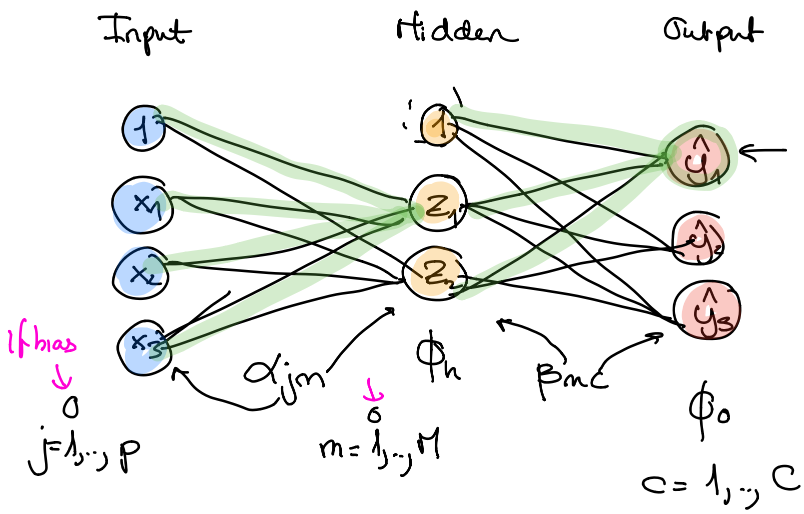

$$\hat{\beta} = (X^T X)^{-1} X^T Y$$

Minimize mean squared loss

$$\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_1(x_i))^2$$

to estimate weights using gradient descent

# Feedforward neural networks




This is a 3-2-3 network (with bias nodes for all layers), and has predicted value for output node c:

$$\hat{y}_c(x) = \phi_o\left(\beta_{oc} + \sum_{m=1}^{M} \beta_{mc} \cdot \phi_h\left(\alpha_{om} + \sum_{j=1}^{p} \alpha_{jm} \cdot x_j\right)\right)$$

How many parameters to estimate?

$\alpha$'s from input to hidden layer $(3+1) \cdot 2 = 8$

$\beta$'s from hidden to output layer $(2+1) \cdot 3 = 9$

} 17 par. in total

* What decides p and C?

C: regression = # response, classification = # classes

p: all covariates we want to use

Regression: $C = 1$, $\phi_o$ usually $\phi_o(a) = a$ identity

Classification $C = 2$, $\phi_o(a) = \dfrac{1}{1+\exp(-a)}$ sigmoid

$C > 2$, $\phi(a) = \dfrac{\exp(a_j)}{\sum_{s=1}^{C} \exp(a_s)}$ softmax

$a_1, a_2, .., a_C$

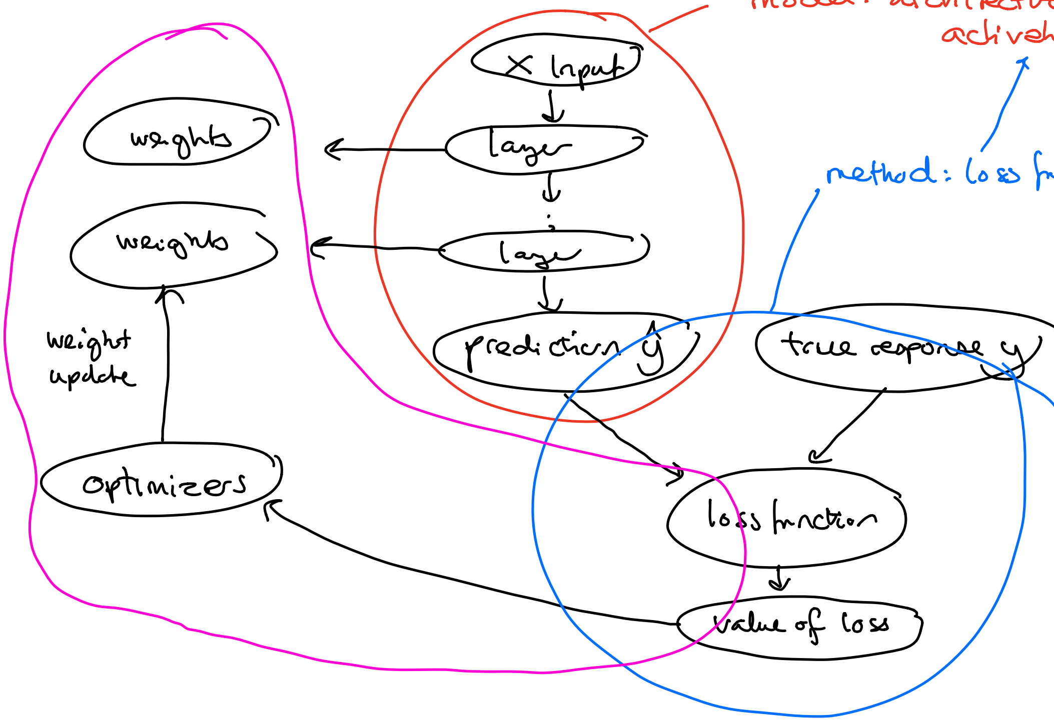
# NEURAL NETWORK PARTS

\# hidden layer & nodes: ?

hidden: ReLU $\phi(a) = \max(0,a)$

model: architecture and activation function

method: loss function




optimization: mini-batch stochastic gradient descent

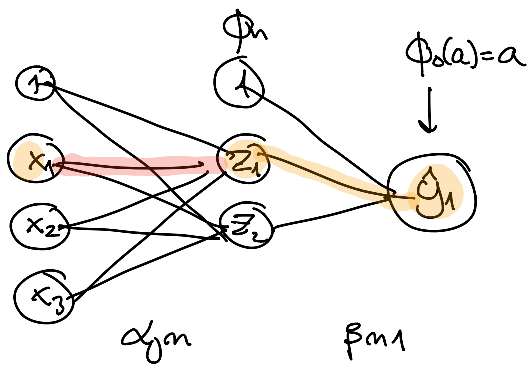
performed by back-propagation

tune learning rate

RMSprop

Regularization
- L1, L2 (weight decay)
- drop-out
- early stopping

Why do we need backpropagation?




Regression 3-2-1 net.

$$\theta = \begin{bmatrix} \alpha_{01} \\ \alpha_{11} \\ \alpha_{12} \\ \vdots \\ \beta_{01} \\ \beta_{11} \\ \beta_{21} \end{bmatrix}$$

$$J(\theta) = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \hat{y}_1(x_i)\right)^2$$

$$\hat{y}_1(x_i) = \beta_{01} + \beta_{11} \cdot z_1 + \beta_{21} \cdot z_2$$

$$z_1 = \phi_h\Big(\underbrace{\alpha_{01} + \alpha_{11}\, x_{i1} + \alpha_{21} \cdot x_{i2} + \alpha_{31} \cdot x_{i3}}_{z_1^*}\Big)$$

$$\frac{\partial J}{\partial \theta} = \begin{bmatrix} \frac{\partial J}{\partial \alpha_{01}} \\ \vdots \\ \frac{\partial J}{\partial \beta_{21}} \end{bmatrix}$$

$$\frac{\partial J}{\partial \alpha_{11}} = \frac{\partial J}{\partial \hat{y}_1} \cdot \frac{\partial \hat{y}_1}{\partial z_1} \cdot \frac{\partial z_1}{\partial \alpha_{11}}$$

$$\frac{\partial z_1}{\partial z_1^*} \frac{\partial z_1^*}{\partial \alpha_{11}}$$

$$-\frac{n}{2} \sum_{i=1}^{n} \left(y_i - \hat{y}_1(x_i)\right) \left[ \beta_{11} \cdot \phi_h'(z_1^*) \cdot x_{i1} \right]$$