Linear models

Neural network $f: \mathbb{R}^{m \times n} \to \mathbb{R}^m$

m – batch size

n – # features

Linear model (single neural unit) — perceptron

$$\vec{y} = f(X) = X\vec{w} + b \qquad w \in \mathbb{R}^n, \; b \in \mathbb{R}$$

elementwise

$$y_i = \sum_{j=1}^{n} w_j x_{ij} + b \qquad (i = 1, 2, \dots, m)$$

Logistic regression

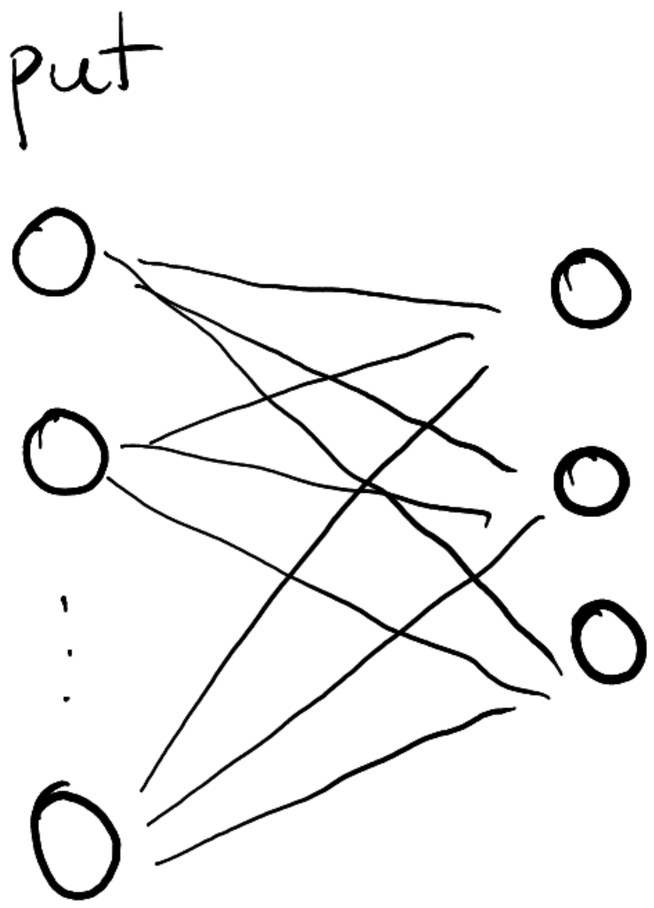
$$\vec{y} = f(X) = \sigma(Xw + b) \qquad \sigma: \mathbb{R}^m \to \mathbb{R}^m$$

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

elementwise

$$y_i = \sigma\left(\sum_{j=1}^{n} w_j x_{ij} + b\right)$$

More neural units

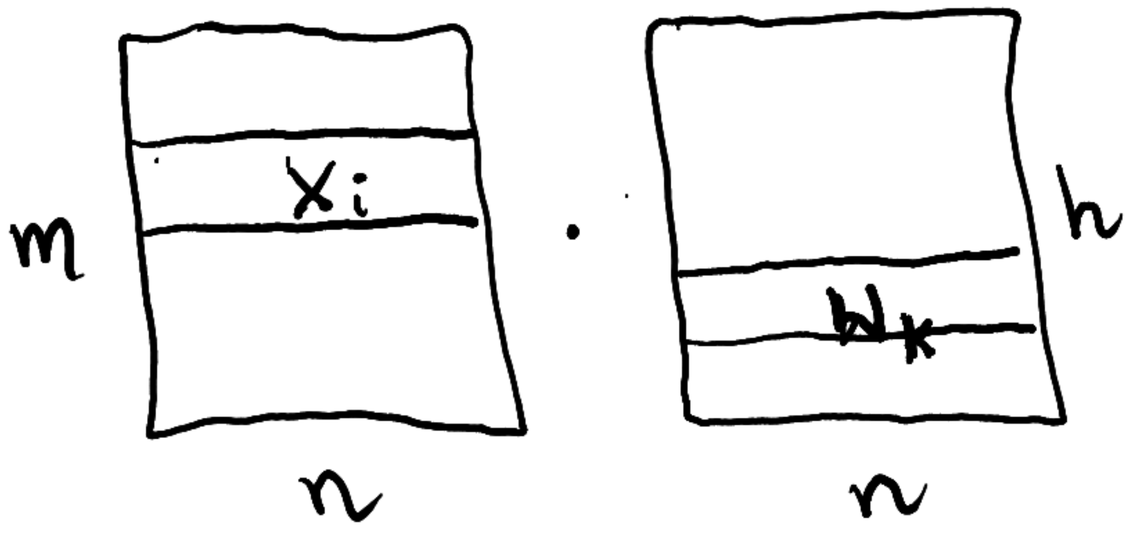


$$Y \in \mathbb{R}^{m \times h}$$

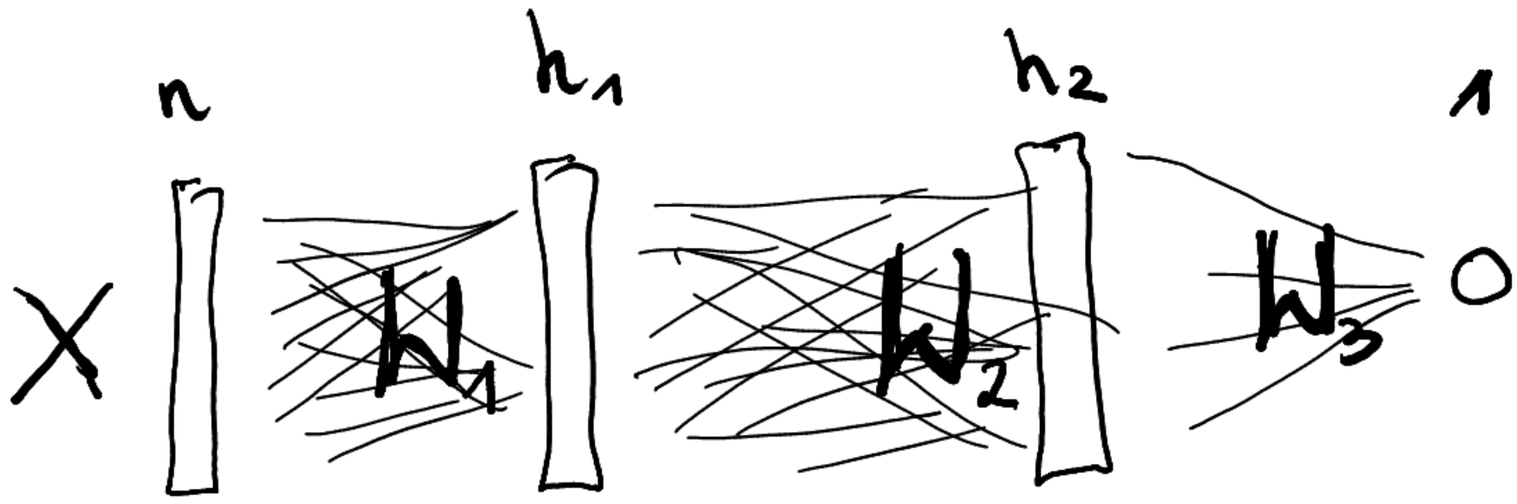
$$X \in \mathbb{R}^{m \times n} \qquad W \in \mathbb{R}^{h \times n} \qquad \vec{b} \in \mathbb{R}^{h}$$

$$Y = f(X) = X W^{T} + b$$

$$Y_{ik} = \sum_{j=1}^{n} x_{ij} \cdot w_{kj} + b_k \qquad \begin{pmatrix} 1 \le i \le m \\ 1 \le k \le h \end{pmatrix}$$



2 - hidden layers + 1 output layer



$X \in \mathbb{R}^{m \times n}$ $W_1 \in \mathbb{R}^{h_1 \times n}$ $W_2 \in \mathbb{R}^{h_2 \times h_1}$ $W_3 \in \mathbb{R}^{1 \times h_2}$

$b_1 \in \mathbb{R}^{h_1}$ $b_2 \in \mathbb{R}^{h_2}$ $b_3 \in \mathbb{R}$

Feed - forword

$$Z_1 = X W_1^T + b_1 \in \mathbb{R}^{m \times h_1}$$

$$A_1 = g(Z_1) \qquad (\text{e.g. } g = \sigma)$$

$$Z_2 = A_1 W_2^T + b_2 \in \mathbb{R}^{m \times h_2}$$

$$A_2 = g(Z_2)$$

$$Z_3 = A_2 W_3^T + b_3 \in \mathbb{R}^{m \times h_3}$$

$$A_3 = g(Z_3)$$

$$y = A_3$$

Pytorch notation

Linear layer

model = torch.nn.Linear(in, out)

- $W \in \mathbb{R}^{out \times in}$
- $b \in \mathbb{R}^{out}$

$model(X) = XW^T + b$

$X \in \mathbb{R}^{m \times in}$

$XW^T \in \mathbb{R}^{m \times out}$

$b \in \mathbb{R}^{out} \rightsquigarrow \mathbb{R}^{m \times out}$

broadcasting

$\mathbb{R}^{m \times out} \oplus \mathbb{R}^{out}$