

Fully Connected Neural Networks

Roberto Souza
Assistant Professor
Electrical and Computer Engineering
Schulich School of Engineering

January 2021

Outline

- Learning Goals
- Fully Connected Neural Networks
 - Single-layer Network
 - Multi-layer Network
- Avoiding Overfitting with Dropout
- Summary

Learning Goals

- Introduce fully connected neural networks
- Learn how to compute the number of parameters of your model
- Learn how to use dropout to avoid model overfitting

Notation

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1M} \\ x_{21} & x_{22} & \dots & x_{2M} \\ x_{31} & x_{32} & \dots & x_{3M} \\ \dots & \dots & \dots & \dots \\ x_{N1} & x_{N2} & \dots & x_{NM} \end{bmatrix}$$

N samples with M
features

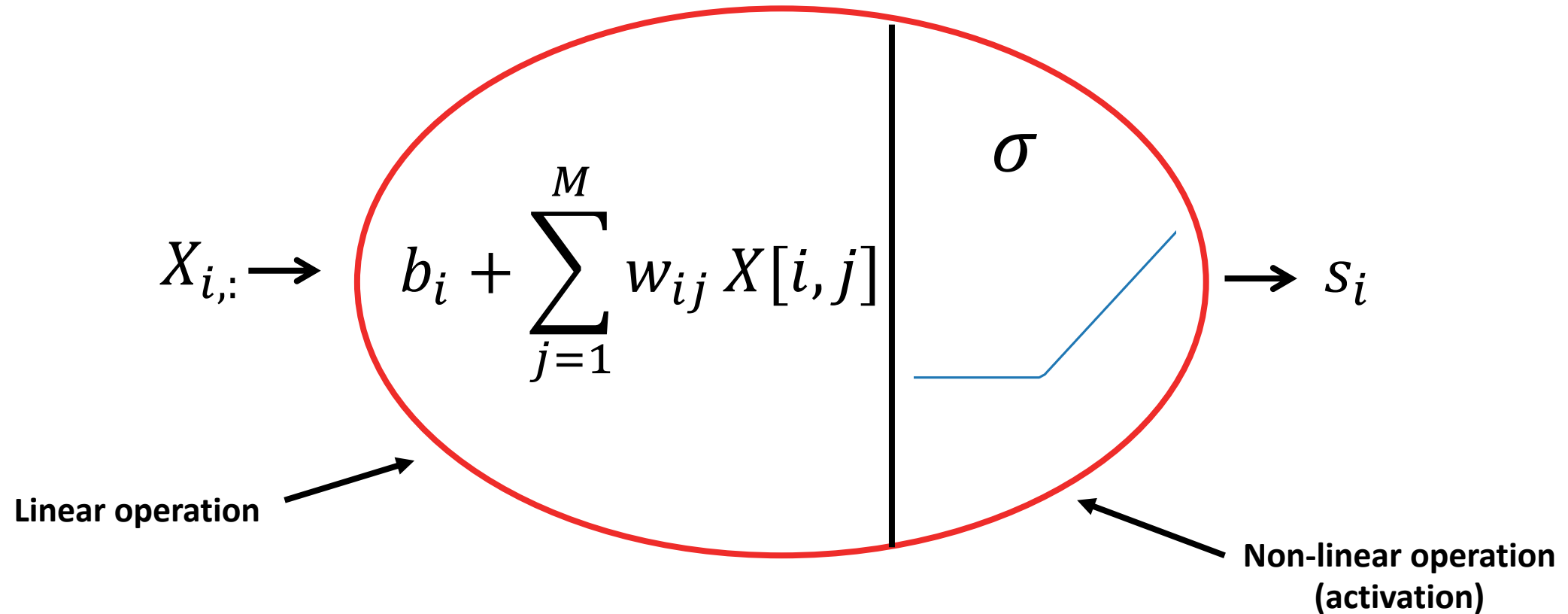
$$Y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \dots \\ y_5 \end{bmatrix}$$

True Labels

$$\hat{Y} = \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \hat{y}_3 \\ \dots \\ \hat{y}_N \end{bmatrix}$$

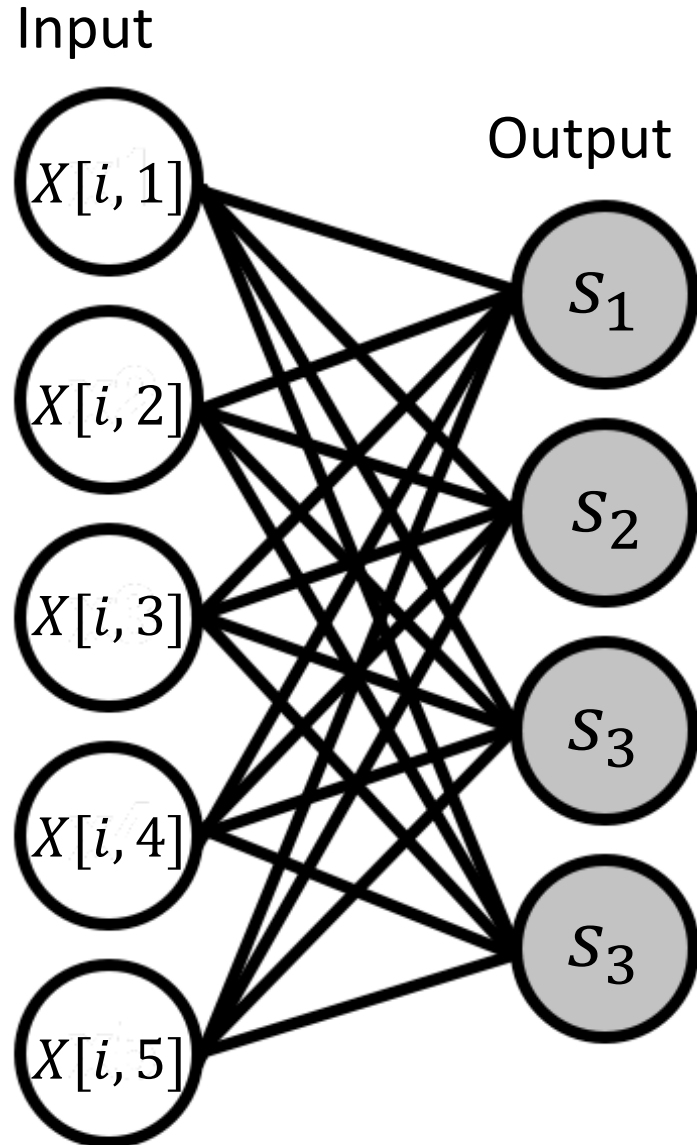
Predicted Labels

The Neuron Model



- b_i is the bias
- w_{ij} are the weights
- s_i is the output of the neuron
- σ is the activation function

Single-layer Fully Connected Neural Network



$$[S]_{C \times 1} = \sigma([W]_{C \times M} X_{i,:}^T + [B]_{C \times 1})$$

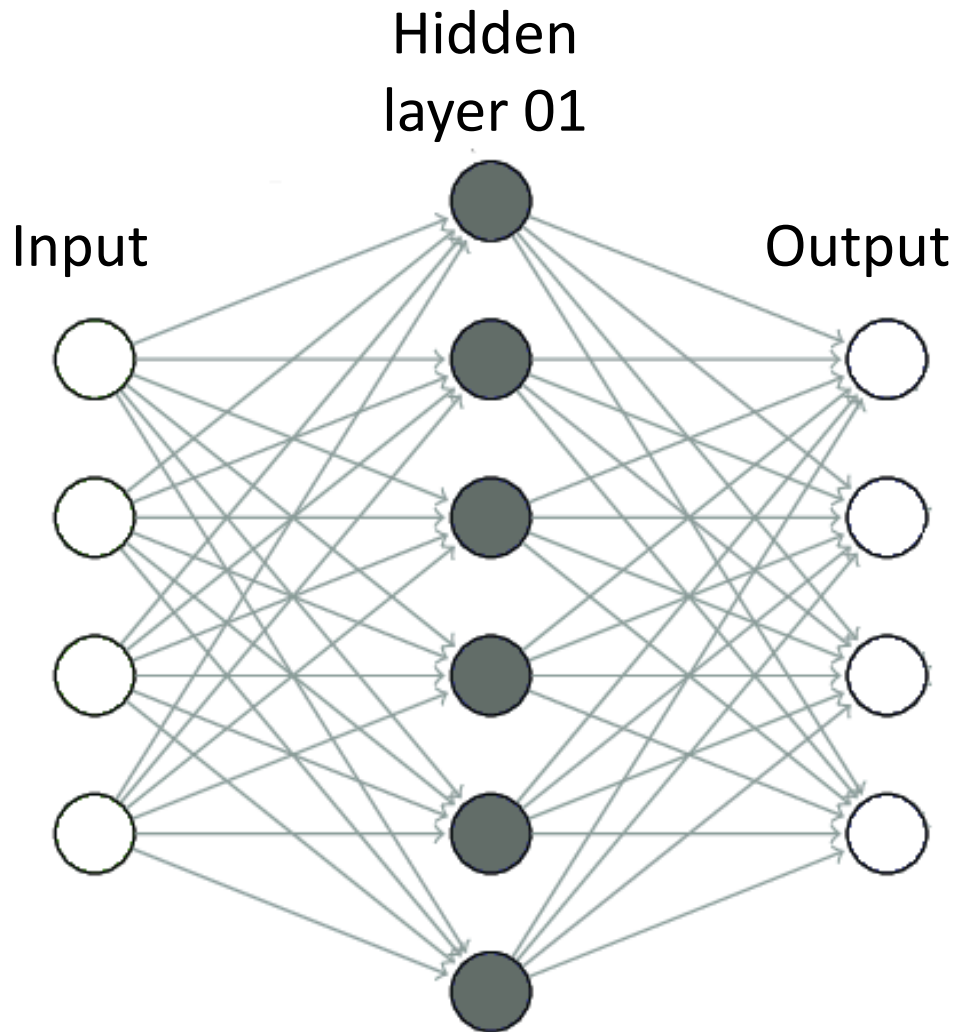
Matrix formulation

If the activation σ is the softmax function, then:

$$\hat{y} = \underset{\forall i}{\operatorname{argmax}}(s_i)$$

- **Number of parameters: $C \times (M+1) = 4 \times 6 = 24$**

Multi-layer Fully Connected Neural Network



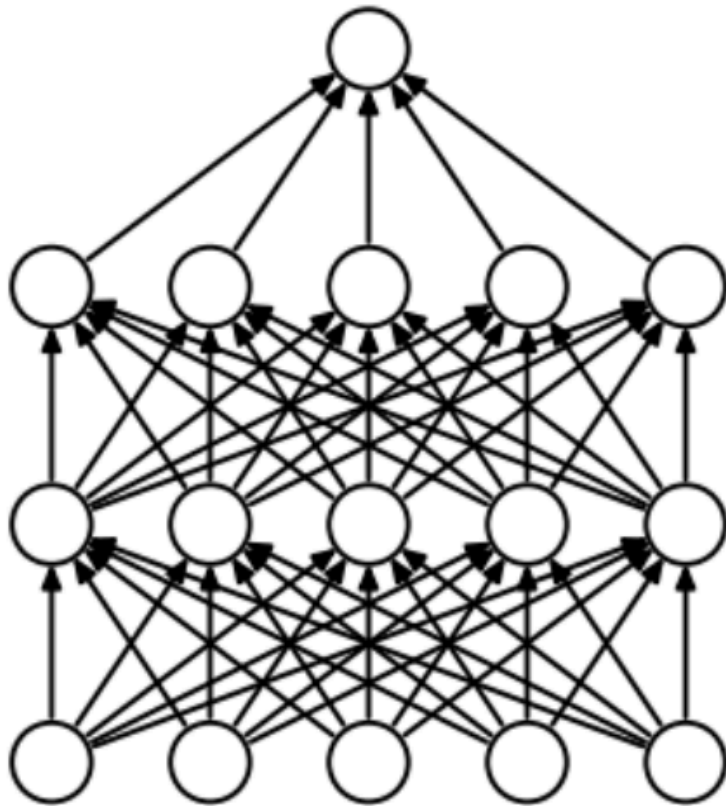
$$[S^{(1)}] = \sigma_1([W^{(1)}] X_{i,:}^T + [B^{(1)}])$$

$$[S^{(2)}] = \sigma_2([W^{(2)}] S^{(1)} + [B^{(2)}])$$

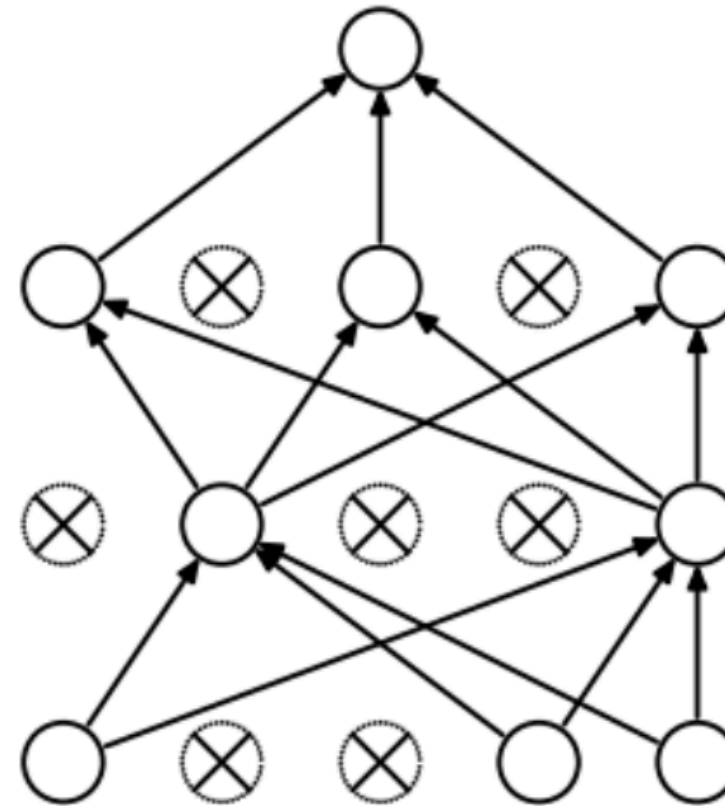
- **Number of parameters:**
 - **First layer: $(4 \times 6) + 6 = 30$**
 - **Second layer: $(6 \times 4) + 4 = 28$**

Dropout

- Technique to avoid overfitting
- Learn redundant paths -> gain robustness



(a) Standard Neural Net



(b) After applying dropout.

Summary

- Fully connected neural networks alternate linear operations (matrix multiplication + bias term) and non-linear activations
- The number of parameters in each layer is given by the (number of inputs +1) x the number of outputs
- Dropout is a technique to avoid overfitting that makes the neural network learn redundant paths to reach the same decision

Thank you!



UNIVERSITY OF
CALGARY

