CS 4700: Foundations of Artificial Intelligence

Spring 2020
Prof. Haym Hirsh

Lecture 29
April 22, 2020

ONLINE

Backup plans:

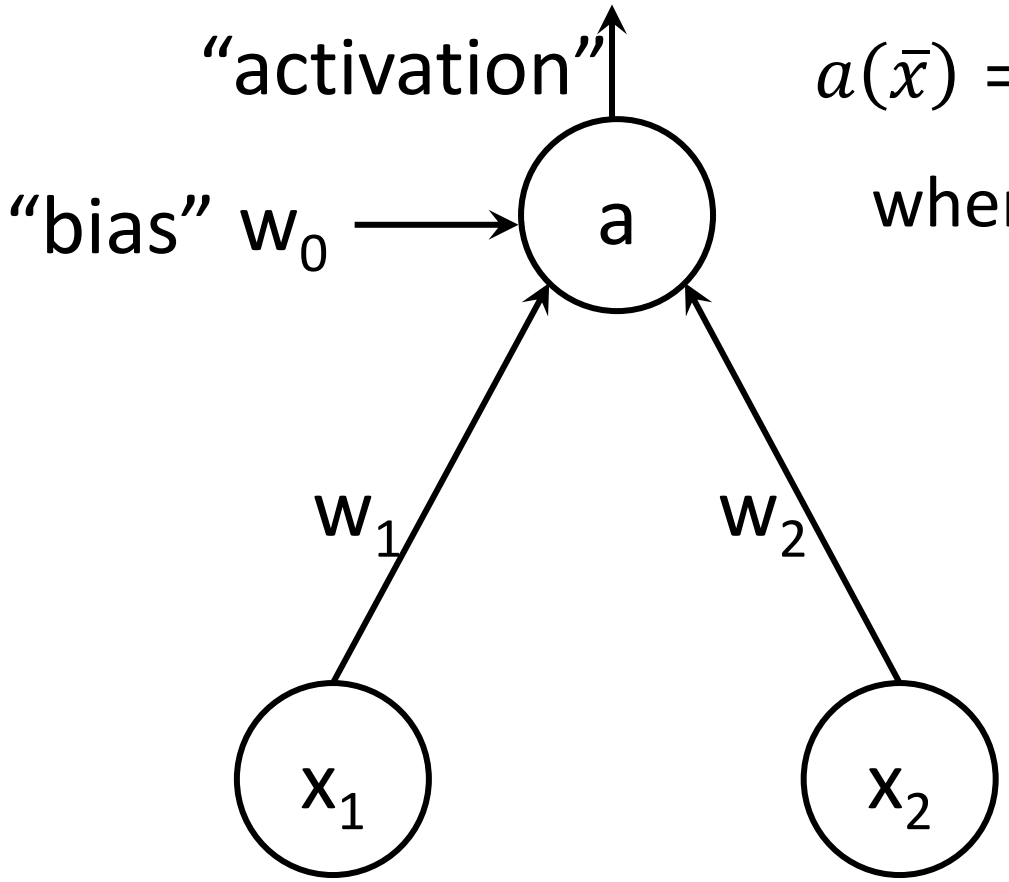
If this Zoom meeting ends prematurely,
five-minute break, check Piazza

Karma Lectures

- Tuesday, April 21, 11:40am: Lily Weng, MIT
  "Evaluating Robustness of Neural Networks"
  - Recorded: https://vod.video.cornell.edu/media/CS+Colloquium-+Lily+Weng/1_1q6zajy6

- Tuesday, April 28, 11:40am: Ashia Wilson, MIT
  "Variational Perspectives on Machine Learning: Algorithms, Inference, and Fairness?"
  - Live: https://cornell.zoom.us/j/276979194

Abstract Representation of a Logistic Neuron



$$a(\bar{x}) = g(\bar{w} \cdot \bar{x})$$

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

Gradient Descent (Logistic Regression)

Current hypothesis: $\mathrm{h}_{\overline{w}}(\bar{x})$

Initialize $w_0, w_1, w_2, \ldots, w_n$ [commonly: small random weights]

Repeat

For i = 1 to N [for each example]

$\mathrm{h} \leftarrow \mathrm{h}_{\overline{w}}(\overline{\mathrm{x}}_i) = g(\overline{w} \cdot \bar{x}_i)$

For j = 0 to n [for each feature]

$$w_j \leftarrow w_j + \alpha\, g(\overline{w} \cdot \bar{x}_i)(1 - g(\overline{w} \cdot \bar{x}_i))(y_i - g(\overline{w} \cdot \bar{x}_i))x_{ij}$$

Until <stopping condition>

Comparing Update Rules

$$w_j \leftarrow w_j + \alpha x_{ij}(y_i - h)$$

$$w_j \leftarrow w_j + \alpha\, g(\bar{w} \cdot \bar{x}_i)(1 - g(\bar{w} \cdot \bar{x}_i))(y_i - g(\bar{w} \cdot \bar{x}_i))x_{ij}$$

Bigger updates when activation is closer to 0.5

Smaller updates the farther it gets from 0.5

= Tries to keep classifier from being wishy-washy

Gradient Descent (Logistic Regression)

Current hypothesis: $\mathrm{h}_{\overline{w}}(\bar{x})$

Initialize $w_0, w_1, w_2, \ldots, w_n$ [commonly: small random weights]

Repeat

For i = 1 to N [for each example]

$\mathrm{h} \leftarrow \mathrm{h}_{\overline{w}}(\overline{\mathrm{x}}_i) = g(\overline{w} \cdot \bar{x}_i)$

For j = 0 to n [for each feature]

$$w_j \leftarrow w_j + \alpha\, g(\overline{w} \cdot \bar{x}_i)(1 - g(\overline{w} \cdot \bar{x}_i))(y_i - g(\overline{w} \cdot \bar{x}_i))x_{ij}$$

Until <stopping condition>

Gradient Descent (Logistic Regression)

Current hypothesis: $\mathrm{h}_{\overline{w}}(\bar{x})$

Initialize $w_0, w_1, w_2, \ldots, w_n$ [commonly: small random weights]

Repeat

For i = 1 to N [for each example]

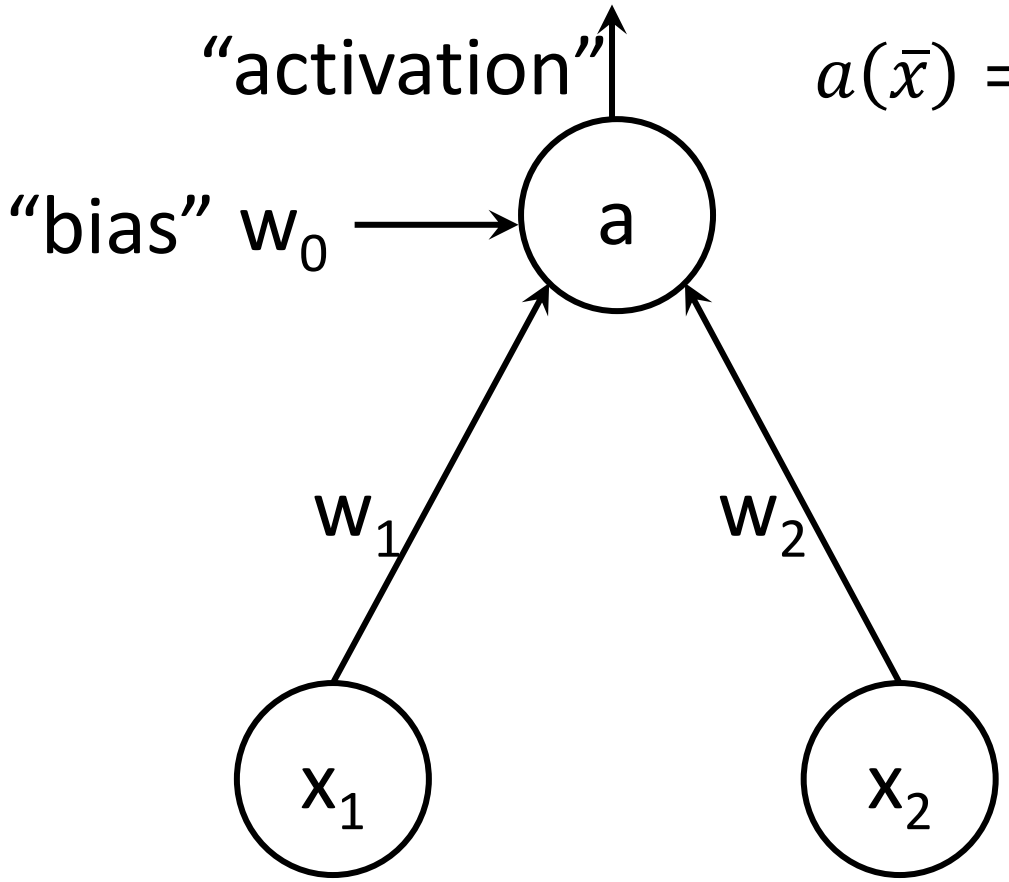
$\mathrm{h} \leftarrow \mathrm{h}_{\overline{w}}(\overline{\mathrm{x}}_i) = g(\overline{w} \cdot \bar{x}_i)$

For j = 0 to n [for each feature]

$\mathrm{w_j} \leftarrow \mathrm{w_j} + \alpha \mathrm{h}(1 - \mathrm{h})(y_i - \mathrm{h})x_{ij}$

Until <stopping condition>

General Case: Abstract Representation of a Neuron



$$a(\bar{x}) = g(\bar{w} \cdot \bar{x})$$

Gradient Descent (Logistic Regression)

Current hypothesis: $\mathrm{h}_{\overline{w}}(\bar{x})$

Initialize $w_0, w_1, w_2, \ldots, w_n$ [commonly: small random weights]

Repeat

For i = 1 to N [for each example]

$\mathrm{h} \leftarrow \mathrm{h}_{\overline{w}}(\overline{\mathrm{x}}_i) = g(\overline{w} \cdot \bar{x}_i)$

For j = 0 to n [for each feature]

$\mathrm{w_j} \leftarrow \mathrm{w_j} + \alpha \mathrm{h}(1 - \mathrm{h})(y_i - \mathrm{h})x_{ij}$

Until <stopping condition>

Gradient Descent: General Case

Current hypothesis: $\mathrm{h}_{\overline{w}}(\bar{x})$

Initialize $w_0, w_1, w_2, \ldots, w_n$ [commonly: small random weights]

Repeat

For i = 1 to N [for each example]

$\mathrm{h} \leftarrow \mathrm{h}_{\overline{w}}(\overline{\mathrm{x}}_i) = g(\overline{w} \cdot \bar{x}_i)$

For j = 0 to n [for each feature]

$w_j \leftarrow w_j + \alpha \mathrm{h}(1 - \mathrm{h})(y_i - \mathrm{h}) x_{ij}$

Until <stopping condition>

Gradient Descent: General Case

Current hypothesis: $\mathrm{h}_{\overline{w}}(\bar{x})$

Initialize $w_0, w_1, w_2, \ldots, w_n$ [commonly: small random weights]

Repeat

For i = 1 to N [for each example]

$\mathrm{h} \leftarrow \mathrm{h}_{\overline{w}}(\overline{\mathrm{x}}_i) = g(\overline{w} \cdot \bar{x}_i)$

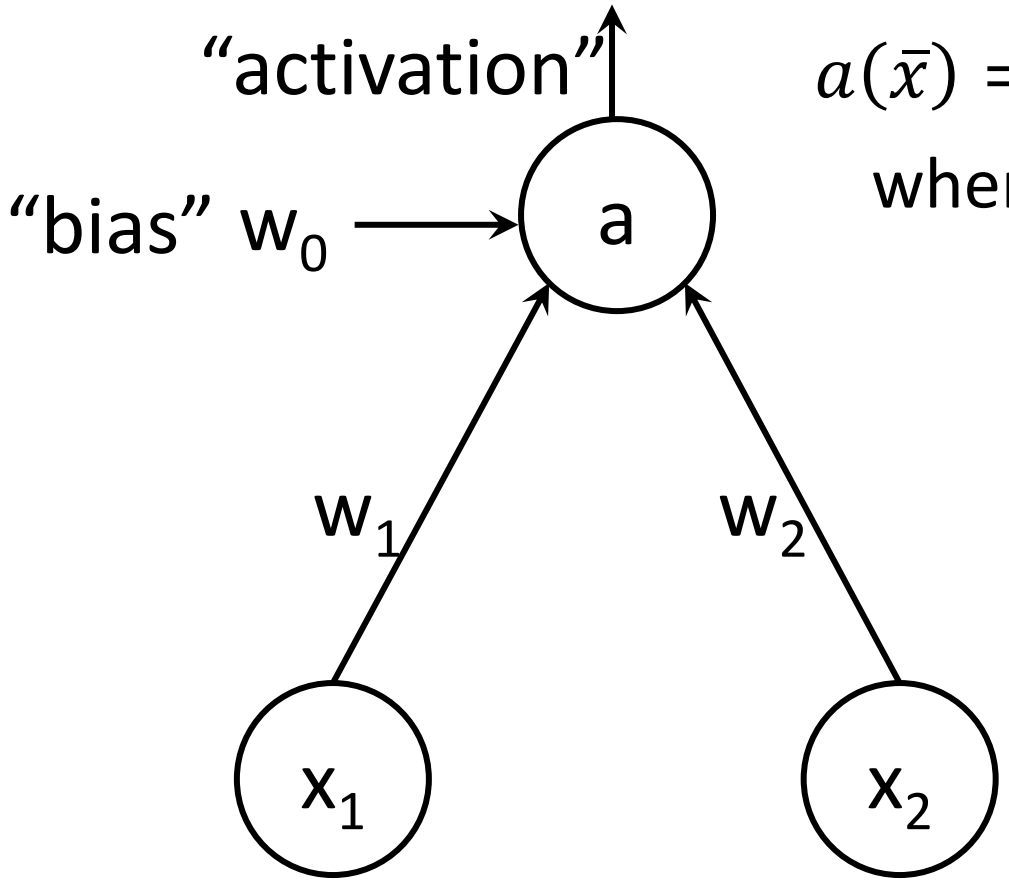
$\mathrm{h}' \leftarrow \mathrm{h}'_{\overline{w}}(\overline{\mathrm{x}}_i) = g'(\overline{w} \cdot \bar{x}_i)$

For j = 0 to n [for each feature]

$$w_j \leftarrow w_j + \alpha \mathrm{h}'(y_i - \mathrm{h}) x_{ij}$$

Until <stopping condition>

Abstract Representation of a tanh Neuron

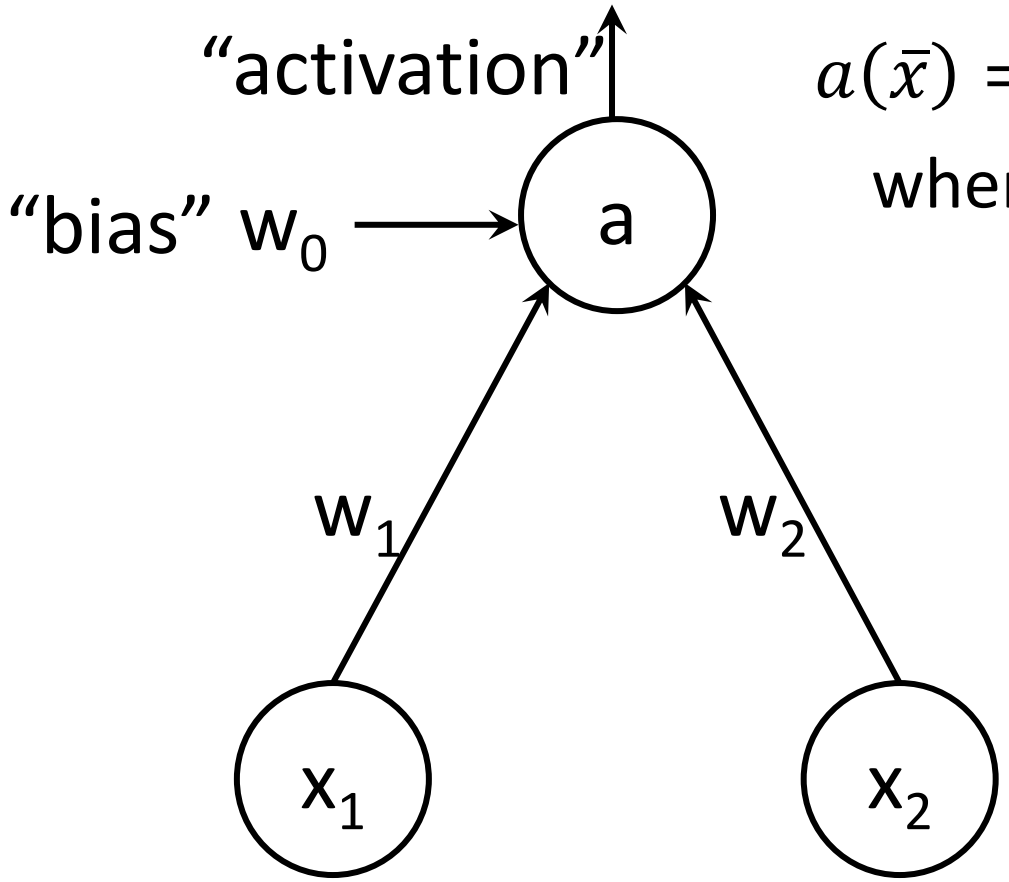


$$a(\bar{x}) = g(\bar{w} \cdot \bar{x})$$

where $g(z) = \tanh(\bar{w} \cdot \bar{x})$

Range (-1,+1)

Abstract Representation of a tanh Neuron



$$a(\bar{x}) = g(\bar{w} \cdot \bar{x})$$

where $g(z) = \tanh(\bar{w} \cdot \bar{x})$

$$g'(z) = 1 - g^2(z)$$

Gradient Descent: General Case

Current hypothesis: $\mathrm{h}_{\overline{w}}(\bar{x})$

Initialize $w_0, w_1, w_2, \ldots, w_n$ [commonly: small random weights]

Repeat

For i = 1 to N [for each example]

$\mathrm{h} \leftarrow \mathrm{h}_{\overline{w}}(\overline{\mathrm{x}}_i) = g(\overline{w} \cdot \bar{x}_i)$

$\mathrm{h}' \leftarrow \mathrm{h}'_{\overline{w}}(\overline{\mathrm{x}}_i) = g'(\overline{w} \cdot \bar{x}_i)$

For j = 0 to n [for each feature]

$$w_j \leftarrow w_j + \alpha \mathrm{h}'(y_i - \mathrm{h}) x_{ij}$$

Until <stopping condition>

Gradient Descent: tanh

Current hypothesis: $\mathrm{h}_{\overline{w}}(\bar{x})$

Initialize $w_0, w_1, w_2, \ldots, w_n$ [commonly: small random weights]

Repeat

For i = 1 to N [for each example]

$\mathrm{h} \leftarrow \mathrm{h}_{\overline{w}}(\overline{\mathrm{x}}_i) = g(\overline{w} \cdot \bar{x}_i) = \tanh(\overline{w} \cdot \bar{x})$

$\mathrm{h}' \leftarrow \mathrm{h}'_{\overline{w}}(\overline{\mathrm{x}}_i) = g'(\overline{w} \cdot \bar{x}_i) = 1 - g^2(z) = 1 - \mathrm{h}^2$

For j = 0 to n [for each feature]

$$w_j \leftarrow w_j + \alpha \mathrm{h}'(y_i - \mathrm{h}) x_{ij}$$

Until <stopping condition>

Gradient Descent: tanh

Current hypothesis: $\mathrm{h}_{\overline{w}}(\bar{x})$

Initialize $w_0, w_1, w_2, \ldots, w_n$ [commonly: small random weights]

Repeat

For i = 1 to N [for each example]

$\mathrm{h} \leftarrow \mathrm{h}_{\overline{w}}(\overline{\mathrm{x}}_i) = g(\overline{w} \cdot \bar{x}_i) = \tanh(\overline{w} \cdot \bar{x})$

For j = 0 to n [for each feature]

$$w_j \leftarrow w_j + \alpha(1 - \mathrm{h}^2)(y_i - \mathrm{h})x_{ij}$$

Until <stopping condition>

Gradient Descent: General Case

Current hypothesis: $\mathrm{h}_{\overline{w}}(\bar{x})$

Initialize $w_0, w_1, w_2, \ldots, w_n$ [commonly: small random weights]

Repeat

For i = 1 to N [for each example]

$\mathrm{h} \leftarrow \mathrm{h}_{\overline{w}}(\overline{\mathrm{x}}_i) = g(\overline{w} \cdot \bar{x}_i)$

$\mathrm{h}' \leftarrow \mathrm{h}'_{\overline{w}}(\overline{\mathrm{x}}_i) = g'(\overline{w} \cdot \bar{x}_i)$

For j = 0 to n [for each feature]

$$w_j \leftarrow w_j + \alpha \mathrm{h}'(y_i - \mathrm{h}) x_{ij}$$

Until <stopping condition>

Gradient Descent: General Case

Current hypothesis: $\mathrm{h}_{\overline{w}}(\bar{x})$

Initialize $w_0, w_1, w_2, \ldots, w_n$ [commonly: small random weights]

Repeat

For i = 1 to N [for each example]

$\mathrm{h} \leftarrow \mathrm{h}_{\overline{w}}(\overline{\mathrm{x}}_i) = g(\overline{w} \cdot \bar{x}_i)$

$\mathrm{h}' \leftarrow \mathrm{h}'_{\overline{w}}(\overline{\mathrm{x}}_i) = g'(\overline{w} \cdot \bar{x}_i)$

For j = 0 to n [for each feature]

$$w_j \leftarrow w_j + \alpha \mathrm{h}'(y_i - \mathrm{h}) x_{ij}$$

Until <stopping condition>

Logistic Regression $\text{Error}_D(\overline{w})$ and Its Gradient

$$\frac{\partial}{\partial wj}\text{Error}_{\text{D}}(\overline{w}) = -2\sum_{i=1}^{N} g'(\overline{w}\cdot\bar{x}_i)(y_i - g(\overline{w}\cdot\bar{x}_i))x_{ij}$$

Focusing on a single example $(\bar{x}_i, y_i)$

$$\frac{\partial}{\partial w_j}\text{Error}_{\{(\bar{x}_i, y_i)\}}(\overline{w}) = -2g'(\overline{w}\cdot\bar{x}_i)(y_i - g(\overline{w}\cdot\bar{x}_i))x_{ij}$$

$$\text{w}_\text{j} \leftarrow \text{w}_\text{j} + \alpha\, g'(\overline{w}\cdot\bar{x}_i)\,(y_i - g(\overline{w}\cdot\bar{x}_i))x_{ij}$$

Logistic Regression $\text{Error}_D(\overline{w})$ and Its Gradient

$$\frac{\partial}{\partial wj}\text{Error}_{\text{D}}(\overline{w}) = -2\sum_{i=1}^{N} g'(\overline{w}\cdot\bar{x}_i)(y_i - g(\overline{w}\cdot\bar{x}_i))x_{ij}$$

Focusing on a single example $(\bar{x}_i, y_i)$

$$\frac{\partial}{\partial w_j}\text{Error}_{\{(\bar{x}_i,y_i)\}}(\overline{w}) = -2g'(\overline{w}\cdot\bar{x}_i)(y_i - g(\overline{w}\cdot\bar{x}_i))x_{ij}$$

$$\text{w}_\text{j} \leftarrow \text{w}_\text{j} + \alpha\, g'(\overline{w}\cdot\bar{x}_i)\,(y_i - g(\overline{w}\cdot\bar{x}_i))x_{ij}$$

This is not the gradient, it’s a single example

Gradient Descent: General Case

Current hypothesis: $\mathrm{h}_{\overline{w}}(\bar{x})$

Initialize $w_0, w_1, w_2, \ldots, w_n$ [commonly: small random weights]

Repeat

For i = 1 to N [for each example]

$\mathrm{h} \leftarrow \mathrm{h}_{\overline{w}}(\overline{\mathrm{x}}_i) = g(\overline{w} \cdot \bar{x}_i)$

$\mathrm{h}' \leftarrow \mathrm{h}'_{\overline{w}}(\overline{\mathrm{x}}_i) = g'(\overline{w} \cdot \bar{x}_i)$

For j = 0 to n [for each feature]

$$w_j \leftarrow w_j + \alpha \mathrm{h}'(y_i - \mathrm{h}) x_{ij}$$

Until <stopping condition>

Gradient Descent: General Case

Current hypothesis: $\mathrm{h}_{\overline{w}}(\bar{x})$

Initialize $w_0, w_1, w_2, \ldots, w_n$ [commonly: small random weights]

Repeat

For i = 1 to N [for each example]

$\mathrm{h} \leftarrow \mathrm{h}_{\overline{w}}(\overline{\mathrm{x}}_i) = g(\overline{w} \cdot \bar{x}_i)$

$\mathrm{h}' \leftarrow \mathrm{h}'_{\overline{w}}(\overline{\mathrm{x}}_i) = g'(\overline{w} \cdot \bar{x}_i)$

For j = 0 to n [for each feature]

$$w_j \leftarrow w_j + \alpha \mathrm{h}'(y_i - \mathrm{h}) x_{ij}$$

[Randomly reorder the data]

Until <stopping condition>

Logistic Regression $\text{Error}_D(\overline{w})$ and Its Gradient

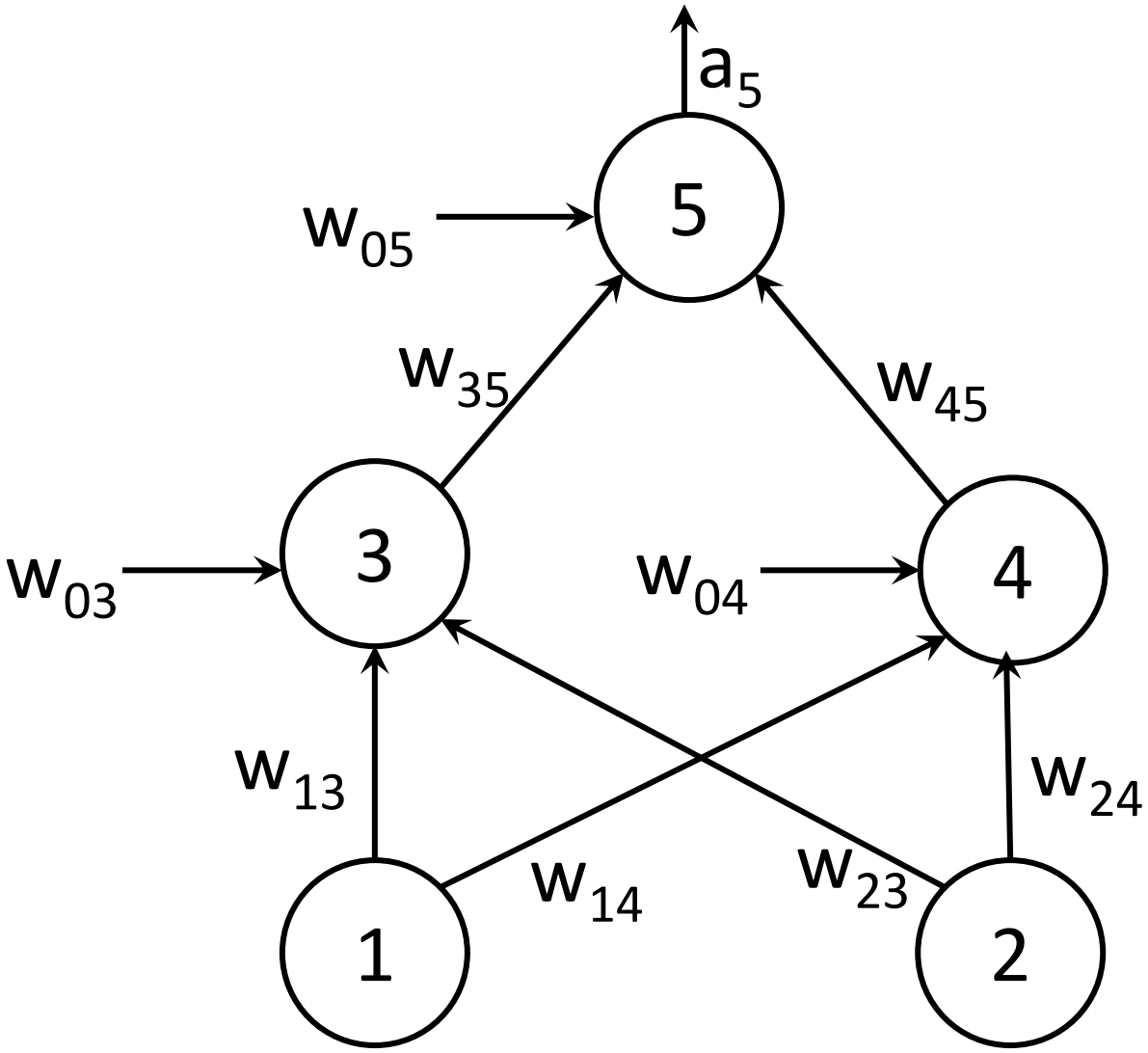
$$\frac{\partial}{\partial wj}\text{Error}_{\text{D}}(\overline{w}) = -2\sum_{i=1}^{N} g'(\overline{w}\cdot\bar{x}_i)(y_i - g(\overline{w}\cdot\bar{x}_i))x_{ij}$$

Focusing on a single example $(\bar{x}_i, y_i)$

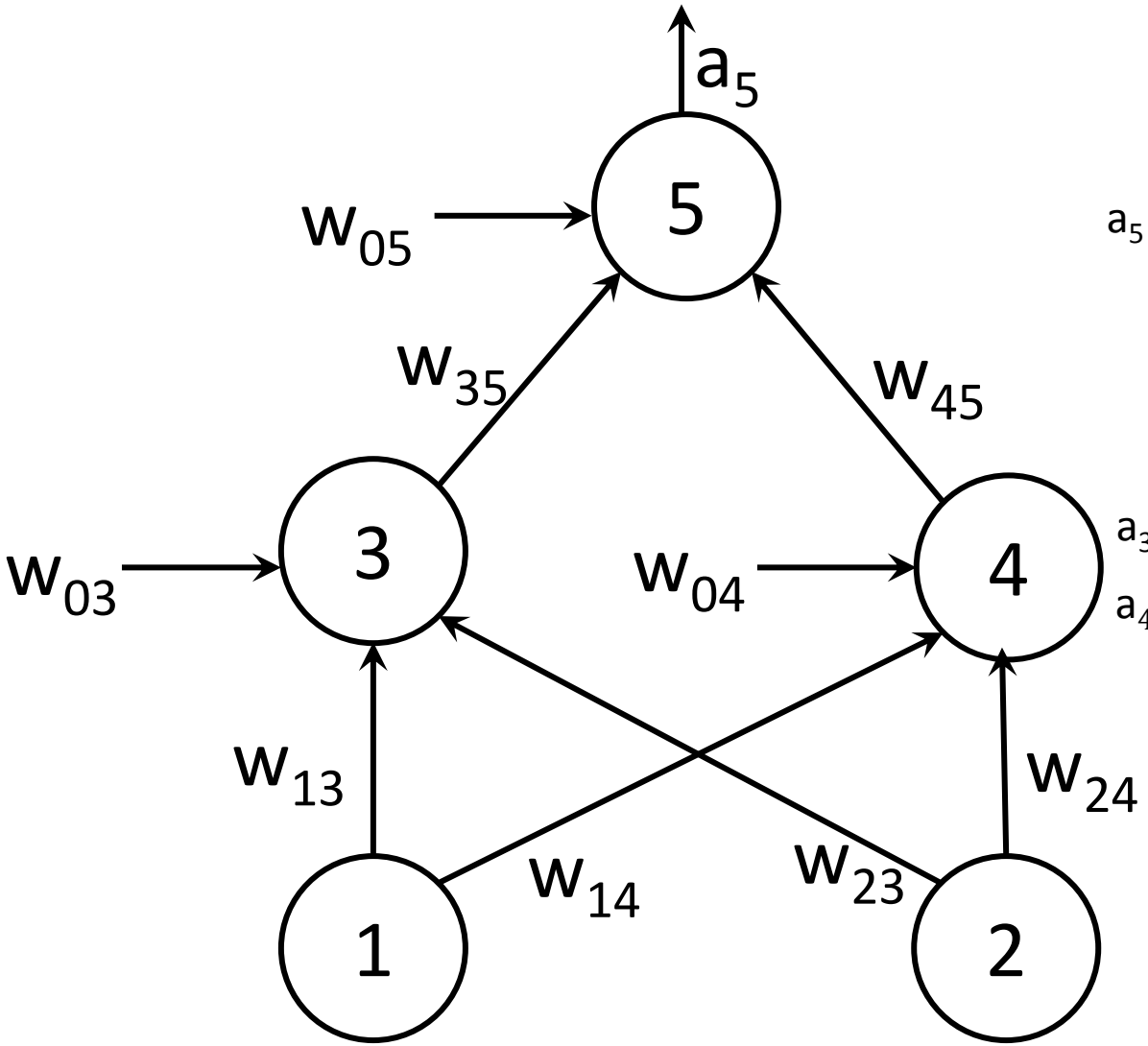
$$\frac{\partial}{\partial w_j}\text{Error}_{\{(\bar{x}_i,y_i)\}}(\overline{w}) = -2g'(\overline{w}\cdot\bar{x}_i)(y_i - g(\overline{w}\cdot\bar{x}_i))x_{ij}$$

Stochastic gradient descent: Updates based on smaller subsets of D

Multi-Layer Neural Network



Multi-Layer Neural Network



$$a_5(\bar{x}_i) = g\big(w_{05} + w_{35}a_3(\bar{x}_i) + w_{45}\,a_4(\bar{x}_i)\big)$$

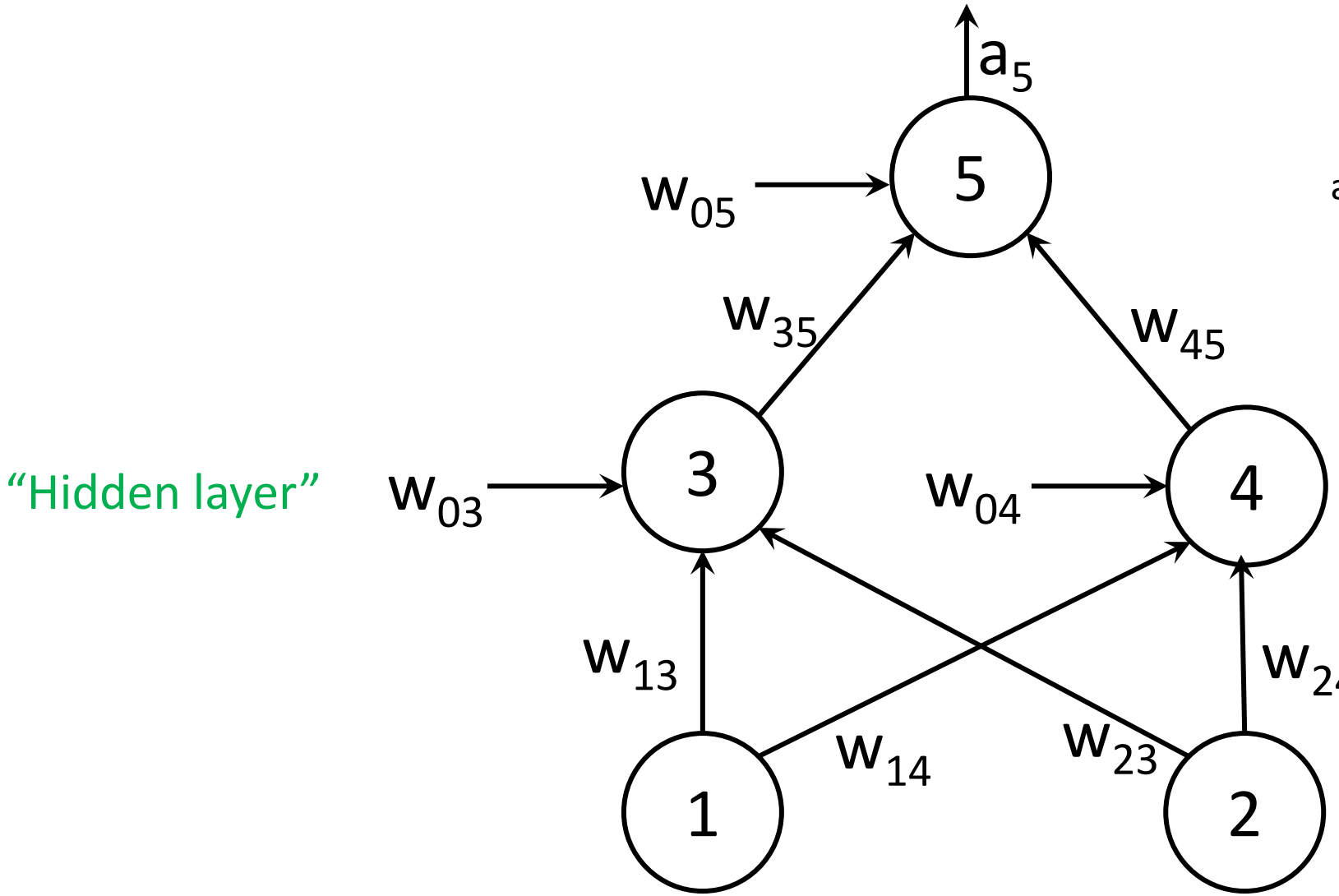
$$a_3(\bar{x}_i) = g\big(w_{03} + w_{13}a_1(\bar{x}_i) + w_{23}\,a_2(\bar{x}_i)\big)$$

$$a_4(\bar{x}_i) = g\big(w_{04} + w_{14}a_1(\bar{x}_i) + w_{24}\,a_2(\bar{x}_i)\big)$$

$$a_1(\bar{x}_i) = x_1$$

$$a_2(\bar{x}_i) = x_2$$

Multi-Layer Neural Network



"Hidden layer"

$$a_5(\bar{x}_i) = g\big(w_{05} + w_{35}a_3(\bar{x}_i) + w_{45}\,a_4(\bar{x}_i)\big)$$

$$a_3(\bar{x}_i) = g\big(w_{03} + w_{13}a_1(\bar{x}_i) + w_{23}\,a_2(\bar{x}_i)\big)$$

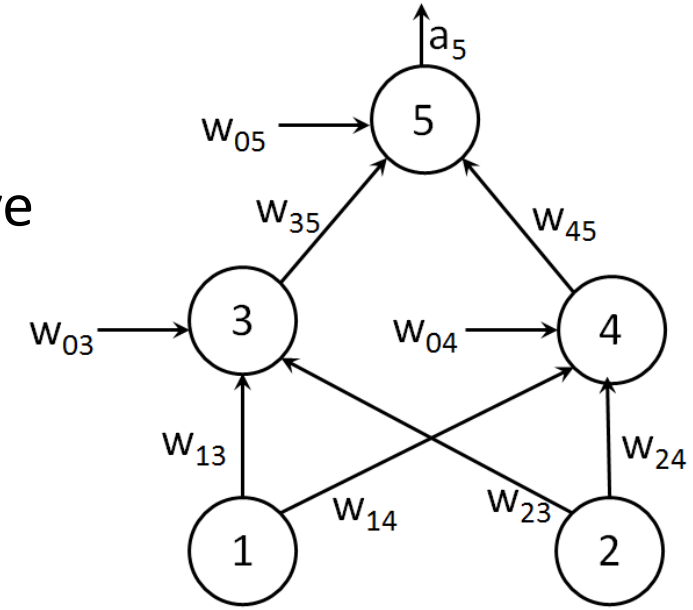
$$a_4(\bar{x}_i) = g\big(w_{04} + w_{14}a_1(\bar{x}_i) + w_{24}\,a_2(\bar{x}_i)\big)$$

$$a_1(\bar{x}_i) = x_1$$

$$a_2(\bar{x}_i) = x_2$$

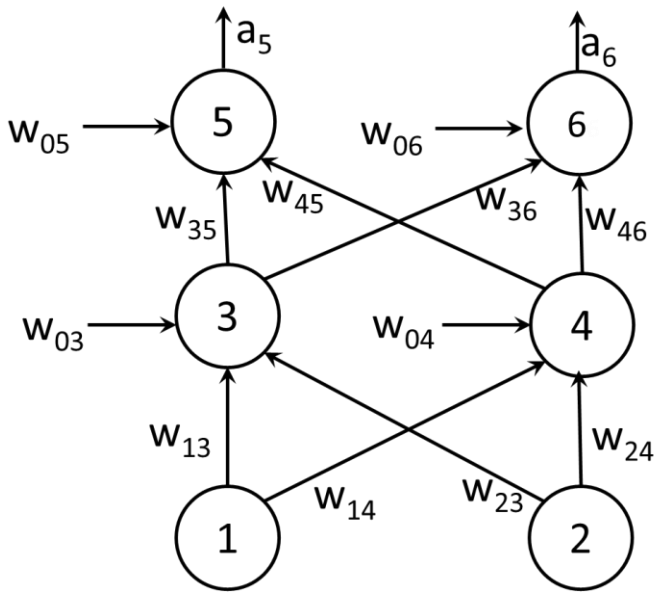
Multi-Layer Neural Network

- Can have more than a single output
  - Example: one output per move in Go – merit of each move
- Can have more than one hidden layer
  - Typically each layer feeds into next layer
- Design choices: # of layers, “width” of layers
- (Many generalizations beyond this)



Multi-Layer Neural Network

- Can have more than a single output
  - Example: one output per move in Go – merit of each move
- Can have more than one hidden layer
  - Typically each layer feeds into next layer
- Design choices: # of layers, “width” of layers
- (Many generalizations beyond this)



Multi-Layer Neural Network

- Increases our “representational power”
- Learning the weights requires something different

Multi-Layer Neural Network

- Increases our "representational power"

- Learning the weights requires something different

  Can compute the derivative of $\frac{\partial}{\partial w_{ij}} \text{Error}_D(\overline{w})$ for each w_{ij}

  Computing $\frac{\partial}{\partial w_{ij}} \text{Error}_D(\overline{w})$ is more complex for lower levels