2-7 Softmax Regression

Zhonglei Wang

WISE and SOE, XMU, 2025

Contents

Introduction

1. Binary classification models
   - $y \in \{0, 1\}$
   - Model is built for $P(y = 1 \mid \boldsymbol{x})$
2. In practice, we may have K classes: $y \in \{0, 1, \ldots, K-1\}$
   - Consider one-hot representation
   - $\boldsymbol{y} = (0, \ldots, 1, \ldots, 0)^{\mathrm{T}}$
   - If $y = k$, only the $(k+1)$th element of $\boldsymbol{y}$ is 1

Model

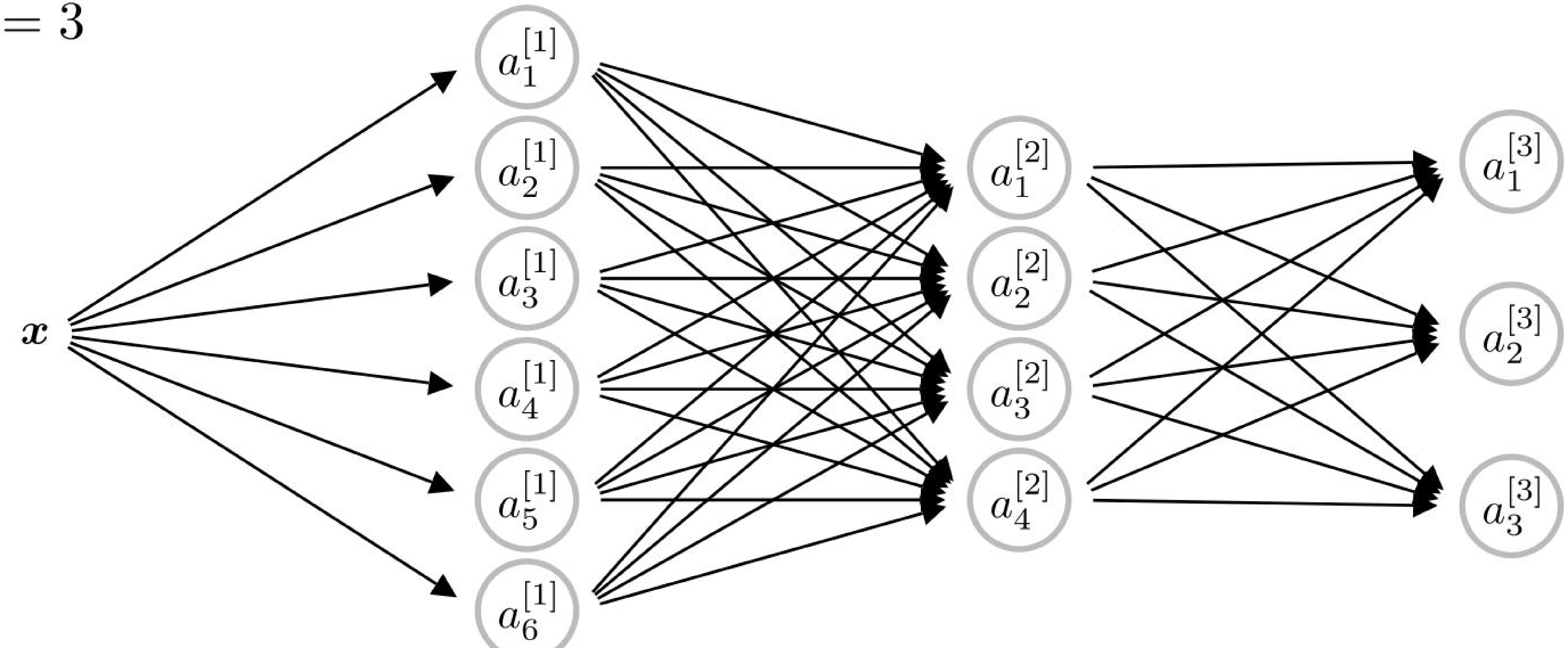
1. For a feature $\boldsymbol{x}$
   - We consider a neural network with K outputs
   - Each output is a score for the corresponding class
   - The scores may not sum up to 1
2. That is, the only difference is the number of neurons in the last layer
   - For binary classification problems, we only have one neuron for the output layer
   - For general classification problems, we have $d^{[L]} = K$ neurons for the output layer

Model

1. Recall that
   - L : number of layers in the neural network
   - $d^{[l]}$: number of neurons in the lth layer $\quad (l = 0, \ldots, L)$
   - $\boldsymbol{a}^{[l]} = (a_1^{[l]}, \ldots, a_{d^{[l]}}^{[l]})^{\mathrm{T}} \in \mathbb{R}^{d^{[l]} \times 1}$
   - $\boldsymbol{W}^{[l]} = (\boldsymbol{w}_1^{[l]}, \ldots, \boldsymbol{w}_{d^{[l]}}^{[l]})^{\mathrm{T}} \in \mathbb{R}^{d^{[l]} \times d^{[l-1]}}$
   - $\boldsymbol{b}^{[l]} = (b_1^{[l]}, \ldots, b_{d^{[l]}}^{[l]})^{\mathrm{T}} \in \mathbb{R}^{d^{[l]} \times 1}$
2. For binary classification problems, we have $d^{[L]} = 1$
3. For general softmax regression problems, we have $d^{[L]} = K$

Model

1. For $K = 3$



Input layer	1st hidden layer	2nd hidden layer	Output layer
$(d^{[0]} = d)$	$(d^{[1]} = 6)$	$(d^{[2]} = 4)$	$(d^{[3]} = 3)$

Forward propogation

1. Let $\boldsymbol{A}^{[0]} = \boldsymbol{X}$
2. For $l = 1, \ldots, L$,

$$\boldsymbol{Z}^{[l]} = \left(\boldsymbol{b}^{[l]}\right)^{\mathrm{T}} + \boldsymbol{A}^{[l-1]} \left(\boldsymbol{W}^{[l]}\right)^{\mathrm{T}}$$
$$\boldsymbol{A}^{[l]} = \sigma^{[l]} \left(\boldsymbol{Z}^{[l]}\right)$$

   - $\sigma^{[l]}(z)$: activation function for the lth layer
   - Broadcasting is used for activation functions

3. For the last layer,

$$\boldsymbol{W}^{[L]} \in \mathbb{R}^{K \times d_{L-1}}, \quad \boldsymbol{b} \in \mathbb{R}^{K \times 1}$$

Forward propogation

1. The cost function for softmax regression is

$$\mathcal{J}(\boldsymbol{\theta}) = -n^{-1} \sum_{i=1}^{n} \sum_{k=1}^{K} y_{ik} \log a_{ik}^{[L]}$$

   - $\boldsymbol{\theta}$: model parameters
   - $\boldsymbol{y}_i = (y_{i1}, \ldots, y_{iK})^{\mathrm{T}}$: one-hot representation for the ith example
   - $a_{ik}^{[L]}$: estimated probablity for the kth class of the ith example

2. The dimension of the $\boldsymbol{A}^{[L]}$, containing estimated probabilities in the last layer
   - $\boldsymbol{A}^{[L]} \in \mathbb{R}^{n\times 1}$ for binary classification problems
   - $\boldsymbol{A}^{[L]} \in \mathbb{R}^{n\times K}$ for softmax regression problems

Backpropogation

1. $\mathrm{d}\boldsymbol{A}^{[L]}$ can be obtained from the cost function
2. Assume $\mathrm{d}\boldsymbol{A}^{[l]}$ is available $\quad (l = L, \ldots, 2)$

$$\mathrm{d}\boldsymbol{Z}^{[l]} = \mathrm{d}\boldsymbol{A}^{[l]} \circ \sigma^{[l]\prime}\left(\boldsymbol{Z}^{[l]}\right)$$

$$\mathrm{d}\boldsymbol{W}^{[l]} = \left(\mathrm{d}\boldsymbol{Z}^{[l]}\right)^{\mathrm{T}} \mathrm{d}\boldsymbol{A}^{[l-1]}$$

$$\mathrm{d}\boldsymbol{b}^{[l]} = \left(\mathrm{d}\boldsymbol{Z}^{[l]}\right)^{\mathrm{T}} \mathbf{1}$$

$$\mathrm{d}\boldsymbol{A}^{[l-1]} = \mathrm{d}\boldsymbol{Z}^{[l]}\boldsymbol{W}^{[l]}$$

3. It remains to obtain $\mathrm{d}\boldsymbol{A}^{[L]}$ for softmax regression

Backpropogation

1. The cost function is

$$\mathcal{J}(\boldsymbol{\theta}) = -n^{-1} \sum_{i=1}^{n} \sum_{k=1}^{K} y_{ik} \log a_{ik}^{[L]}$$

2. Thus, the ikth component of $\mathrm{d}\boldsymbol{A}^{[L]} is$

$$\frac{\partial \mathcal{J}}{\partial a_{ik}^{[L]}} = -\frac{y_{ik}}{a_{ik}^{[L]}}$$

3. Done!

Example

Test image

Model (FNN with 2 hidden layers)

Input layer	1st hidden layer	2nd hidden layer	Output layer
$d^{[0]} = 784$	$d^{[1]} = 128$	$d^{[2]} = 64$	$d^{[3]} = 10$

Estimated result

$\longrightarrow$ 2