Week4 Recitation

Zoey Song
CS190I Deep Learning

UC SANTA BARBARA

UC SANTA BARBARA
Computer Science

Outline

- Training & Inference
- Gradient Descent
- Regularization
- Example for Forward & Backward propagation

Classification Task

- E.g., hand-written digit classification

- Training
  - Cross-entropy
    - Model output: $\hat{y}$ = Softmax(Wh+b)
    - Loss: l = -y log $\hat{y}$
- Inference
  - i = $\underset{i \in \{1,2,..,k\}}{\operatorname{argmax}} \hat{y}_i$

Regression Task

- E.g., predict stock price
- Training
  - MSE-loss
    - Model output: $\hat{y}$ = f(Wh+b)
    - Loss: l = ½ (y - $\hat{y}$)^2
- Inference
  - $\hat{y}$ = f(Wh+b)

Gradient Descent

- **Considering the optimization objective**

$$\min_{\theta} F(x, \theta)$$

- Suppose we have N samples, at time step t
  - Full gradient descent

$$\theta^t = \theta^{t-1} - lr_t \frac{1}{N} \sum_{i=1}^{N} \nabla F(x_i, \theta^{t-1})$$

    - Highly Efficient
    - but can not fully utilize the data

Gradient Descent

- Suppose we have N samples, at time step t
  - Stochastic gradient descent
    $\theta^t = \theta^{t-1} - lr_t \nabla F(x_k, \theta^{t-1})$, k in {1, 2, …, N}
    - Computation is slow
    - Can fully utilize the training data
  - Mini-batch with batch size s
    $\theta^t = \theta^{t-1} - lr_t \frac{1}{s} \sum_{k \epsilon t_s} \nabla F(x_k, \theta^{t-1}),\ t_s \sqsubseteq \{1, 2, \ldots, N\}$
    - Trade-off between the computation speed and data use

Regularization

- Prevent overfitting
- L1 regularizer (Lasso regularizer)

$$Loss = Error(y, \hat{y}) + \lambda \sum_{i=1}^{N} |w_i|$$

  - E.g., Alignment in machine translation, graph for social network

- L2 regularizer (Ridge regularizer)

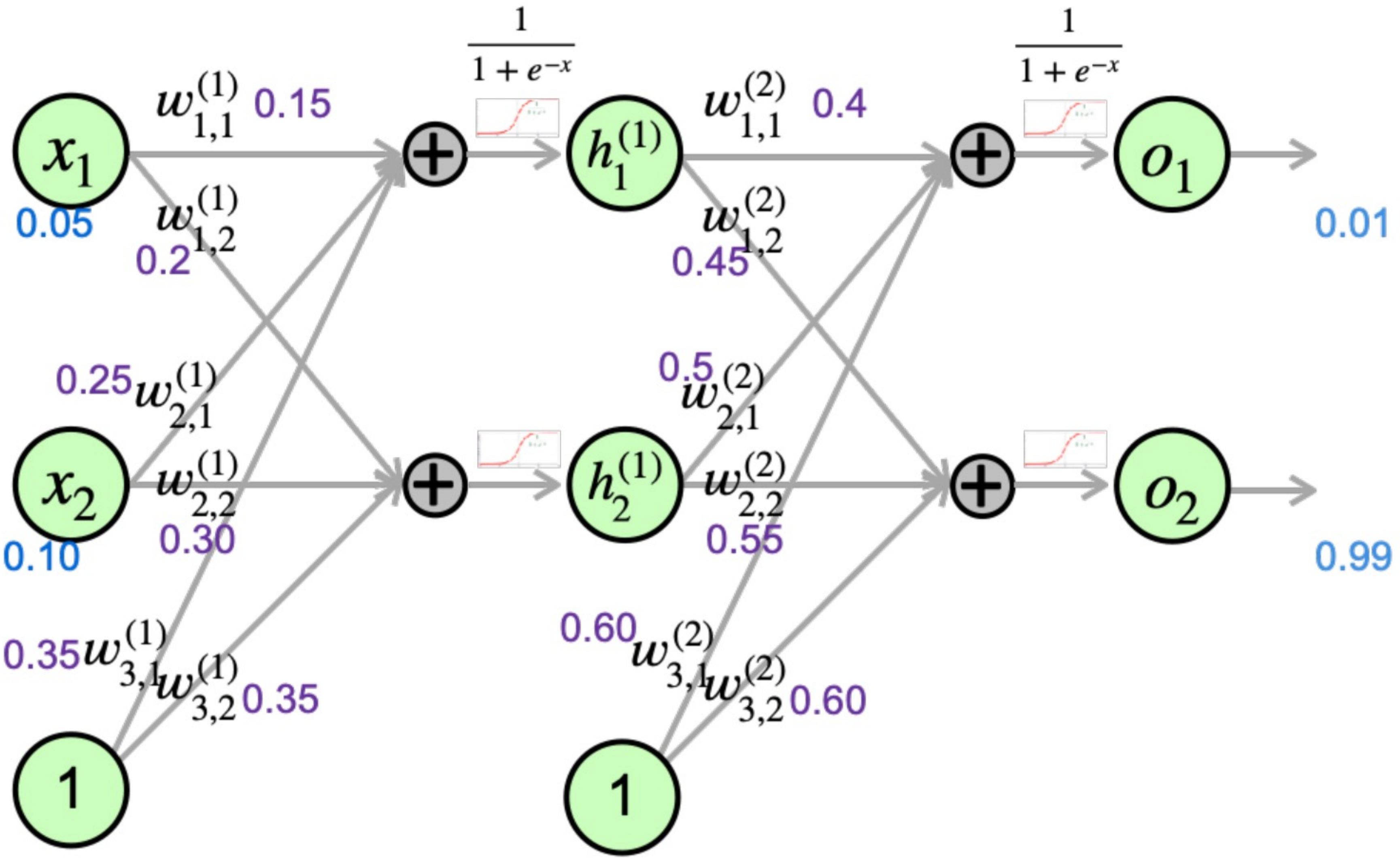
$$J(\theta) = \frac{1}{2m}\left[\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2 + \boxed{\lambda \sum_{j=1}^{n} \theta_j^2}\right]$$

$$\min_\theta J(\theta)$$

  - E.g. deep neural network training

- Dropout
  - 0.1 ~ 0.5

Quiz



$\frac{1}{1+e^{-x}}$
$\frac{1}{1+e^{-x}}$
x_1
0.05
x_2
0.10
1
$w^{(1)}_{1,1}$ 0.15
$w^{(1)}_{1,2}$ 0.2
0.25 $w^{(1)}_{2,1}$
$w^{(1)}_{2,2}$ 0.30
0.35 $w^{(1)}_{3,1}$
$w^{(1)}_{3,2}$ 0.35
$h^{(1)}_1$
$h^{(1)}_2$
1
$w^{(2)}_{1,1}$ 0.4
$w^{(2)}_{1,2}$ 0.45
0.5 $w^{(2)}_{2,1}$
$w^{(2)}_{2,2}$ 0.55
0.60 $w^{(2)}_{3,1}$
$w^{(2)}_{3,2}$ 0.60
o_1
o_2
0.01
0.99

Why calculating gradient is necessary?

- **Sometimes we can't directly compute the gradient**
  - Incorporating a latent variable into the MLE objective

$$\log P(x;\theta) = \log \int P(x,z;\theta)dz$$

Any Question?