# Module 4.3: Output Functions and Loss Functions

### We need to answer two questions

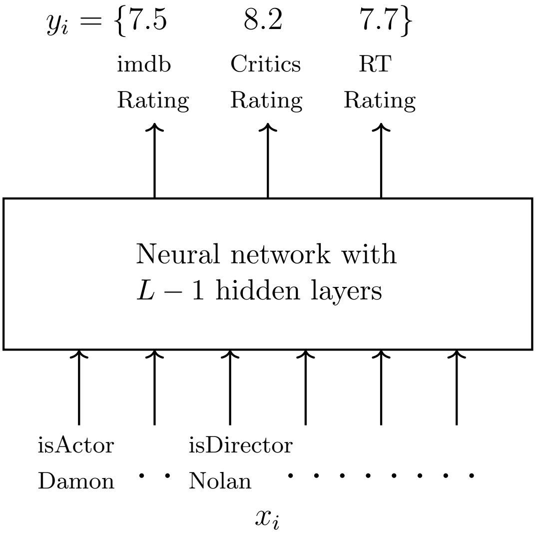
- How to choose the loss function $\mathscr{L}(\theta)$ ?
- How to compute $\nabla\theta$ which is composed of $\nabla W_1, \nabla W_2, ..., \nabla W_{L-1} \in \mathbb{R}^{n \times n}, \nabla W_L \in \mathbb{R}^{n \times k}$,
  $\nabla b_1, \nabla b_2, ..., \nabla b_{L-1} \in \mathbb{R}^n$ and $\nabla b_L \in \mathbb{R}^k$ ?

### We need to answer two questions

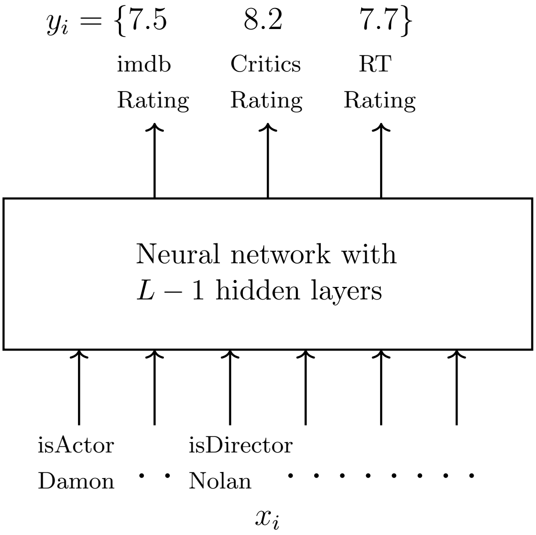
- How to choose the loss function $\mathscr{L}(\theta)$ ?
- How to compute $\nabla\theta$ which is composed of $\nabla W_1, \nabla W_2, ..., \nabla W_{L-1} \in \mathbb{R}^{n\times n}, \nabla W_L \in \mathbb{R}^{n\times k}$,
  $\nabla b_1, \nabla b_2, ..., \nabla b_{L-1} \in \mathbb{R}^n$ and $\nabla b_L \in \mathbb{R}^k$ ?

- The choice of loss function depends on the problem at hand

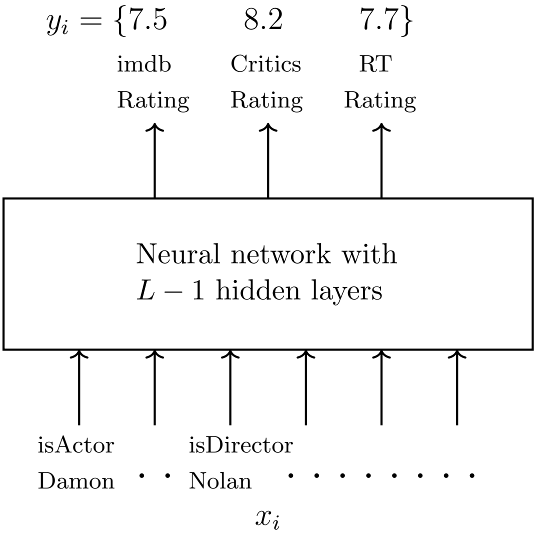
- The choice of loss function depends on the problem at hand
- We will illustrate this with the help of two examples



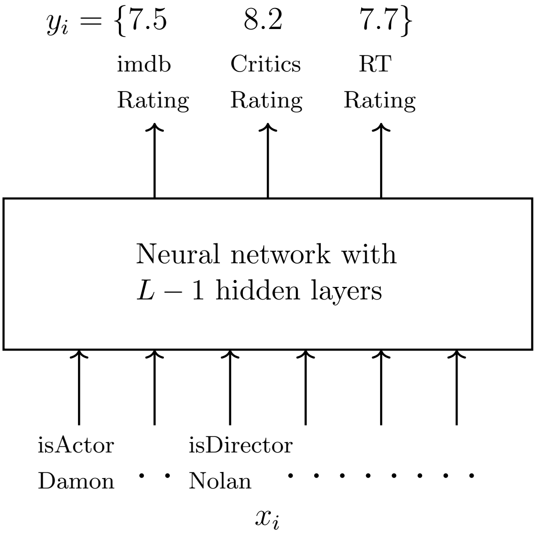

- The choice of loss function depends on the problem at hand
- We will illustrate this with the help of two examples
- Consider our movie example again but this time we are interested in predicting ratings




- The choice of loss function depends on the problem at hand
- We will illustrate this with the help of two examples
- Consider our movie example again but this time we are interested in predicting ratings
- Here $y_i \in \mathbb{R}^3$



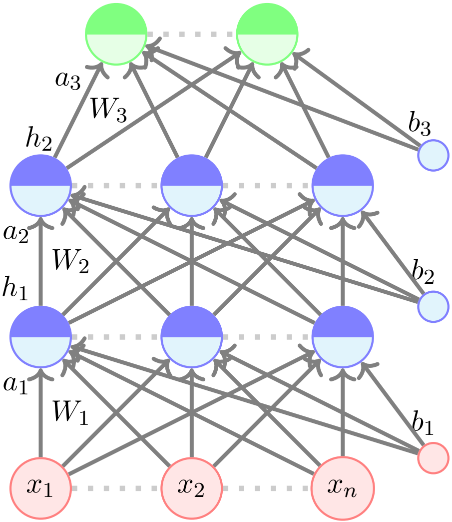

- The choice of loss function depends on the problem at hand
- We will illustrate this with the help of two examples
- Consider our movie example again but this time we are interested in predicting ratings
- Here $y_i \in \mathbb{R}^3$
- The loss function should capture how much $\hat{y}_i$ deviates from $y_i$



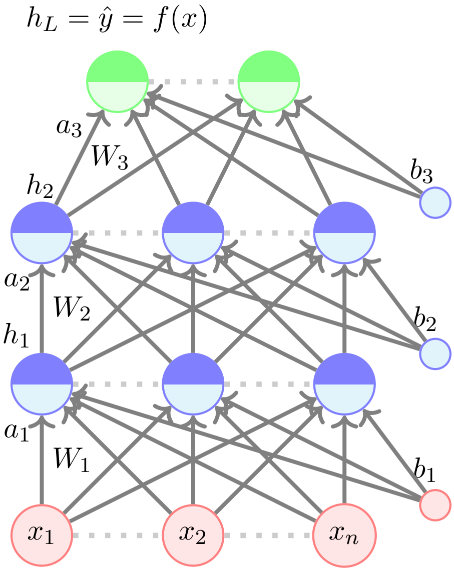

- The choice of loss function depends on the problem at hand
- We will illustrate this with the help of two examples
- Consider our movie example again but this time we are interested in predicting ratings
- Here $y_i \in \mathbb{R}^3$
- The loss function should capture how much $\hat{y}_i$ deviates from $y_i$
- If $y_i \in \mathbb{R}^n$ then the squared error loss can capture this deviation

$$\mathscr{L}(\theta) = \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{3} (\hat{y}_{ij} - y_{ij})^2$$

$h_L = \hat{y} = f(x)$

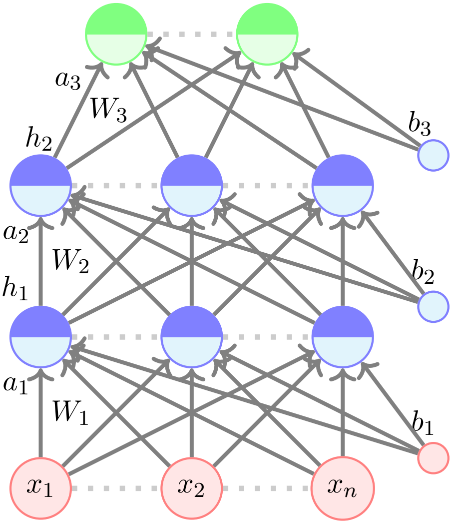



- A related question: What should the output function '$O$' be if $y_i \in \mathbb{R}$?

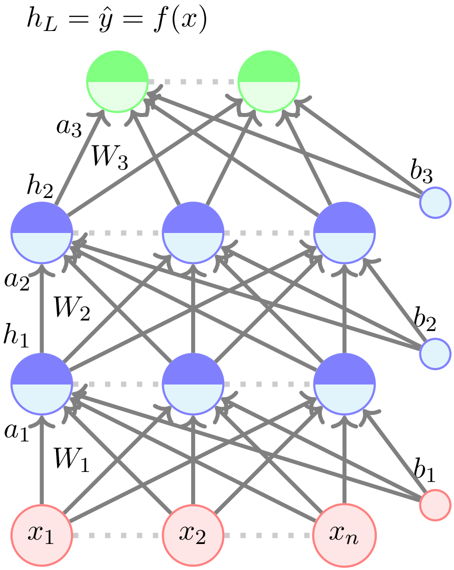



- A related question: What should the output function '$O$' be if $y_i \in \mathbb{R}$?
- More specifically, can it be the logistic function?

$h_L = \hat{y} = f(x)$

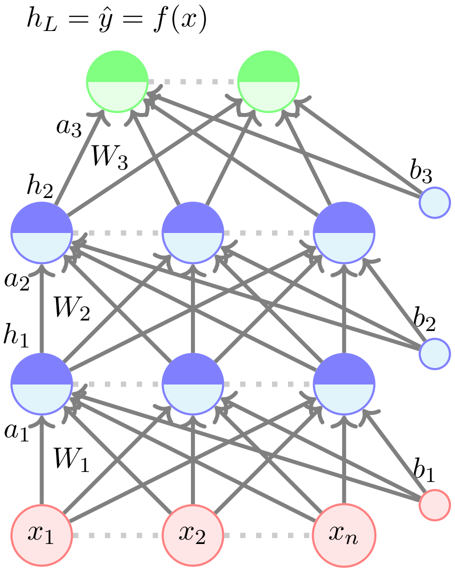



- A related question: What should the output function '$O$' be if $y_i \in \mathbb{R}$?
- More specifically, can it be the logistic function?
- No, because it restricts $\hat{y}_i$ to a value between 0 & 1 but we want $\hat{y}_i \in \mathbb{R}$




- A related question: What should the output function '$O$' be if $y_i \in \mathbb{R}$?
- More specifically, can it be the logistic function?
- No, because it restricts $\hat{y}_i$ to a value between 0 & 1 but we want $\hat{y}_i \in \mathbb{R}$
- So, in such cases it makes sense to have '$O$' as linear function

$$
\begin{aligned}
f(x) &= h_L = O(a_L) \\
&= W_O a_L + b_O
\end{aligned}
$$

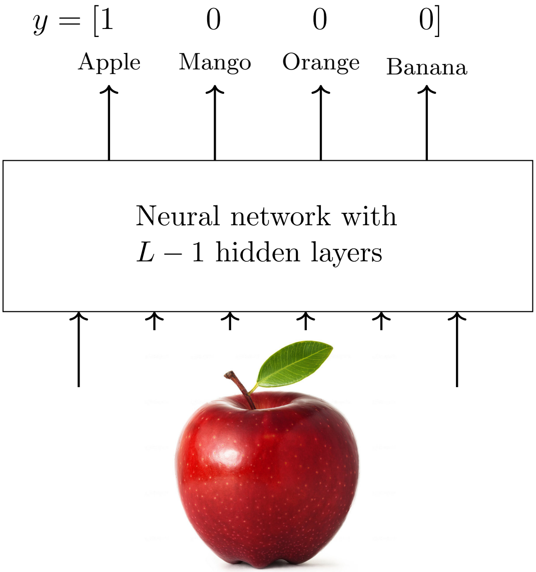



- A related question: What should the output function '$O$' be if $y_i \in \mathbb{R}$?
- More specifically, can it be the logistic function?
- No, because it restricts $\hat{y}_i$ to a value between 0 & 1 but we want $\hat{y}_i \in \mathbb{R}$
- So, in such cases it makes sense to have '$O$' as linear function

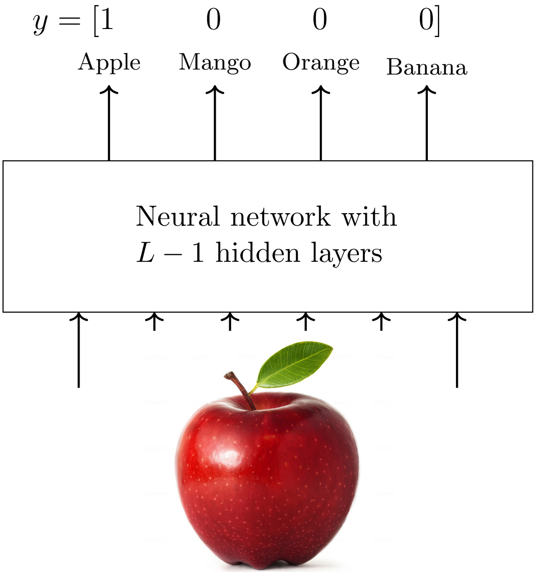
$$\begin{aligned} f(x) &= h_L = O(a_L) \\ &= W_O a_L + b_O \end{aligned}$$

- $\hat{y}_i = f(x_i)$ is no longer bounded between 0 and 1

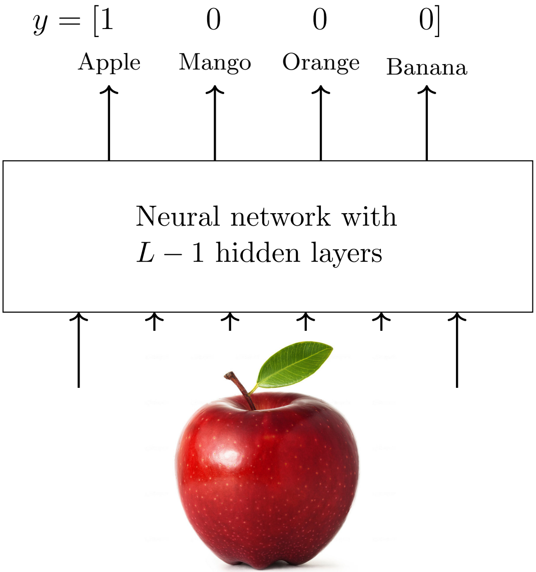
Intentionally left blank



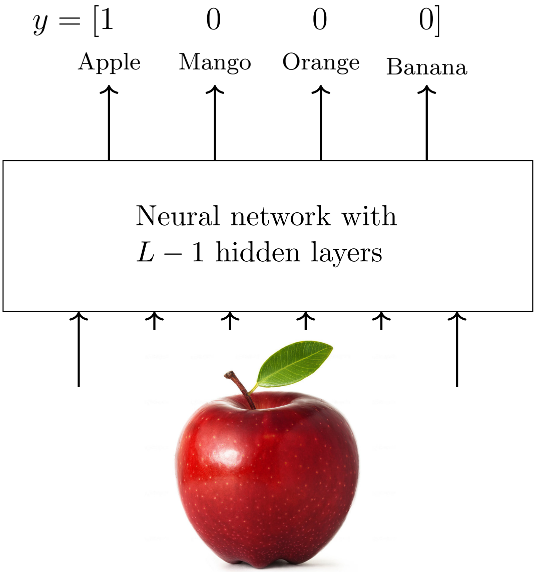

- Now let us consider another problem for which a different loss function would be appropriate




- Now let us consider another problem for which a different loss function would be appropriate
- Suppose we want to classify an image into 1 of $k$ classes




- Now let us consider another problem for which a different loss function would be appropriate
- Suppose we want to classify an image into 1 of $k$ classes
- Here again we could use the squared error loss to capture the deviation



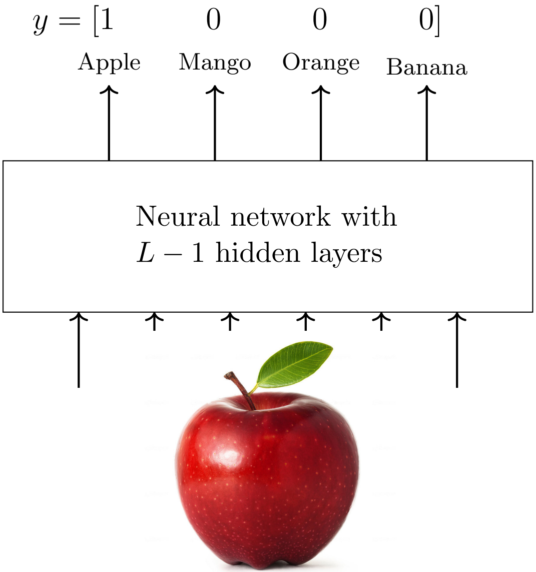

- Now let us consider another problem for which a different loss function would be appropriate
- Suppose we want to classify an image into 1 of $k$ classes
- Here again we could use the squared error loss to capture the deviation
- But can you think of a better function?

$$y = [1 \quad 0 \quad 0 \quad 0]$$

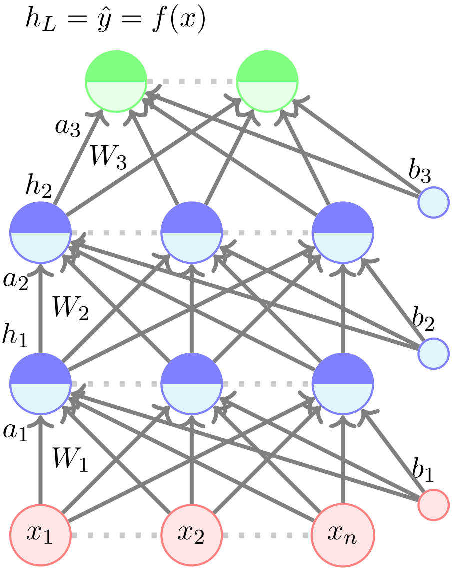
Apple Mango Orange Banana

Neural network with
$L-1$ hidden layers

- Notice that $y$ is a probability distribution

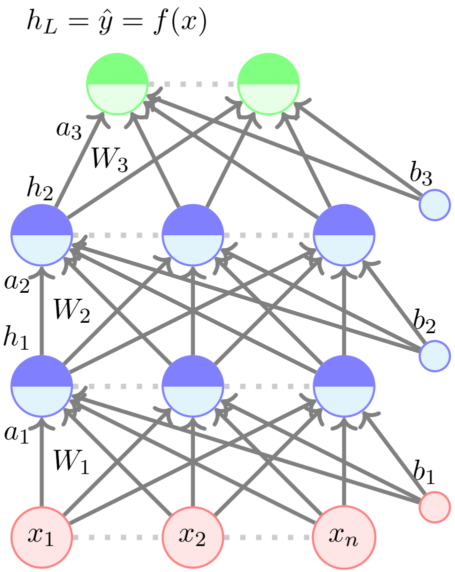



- Notice that $y$ is a probability distribution
- Therefore we should also ensure that $\hat{y}$ is a probability distribution




- Notice that $y$ is a probability distribution
- Therefore we should also ensure that $\hat{y}$ is a probability distribution
- What choice of the output activation '$O$' will ensure this ?

$$a_L = W_L h_{L-1} + b_L$$

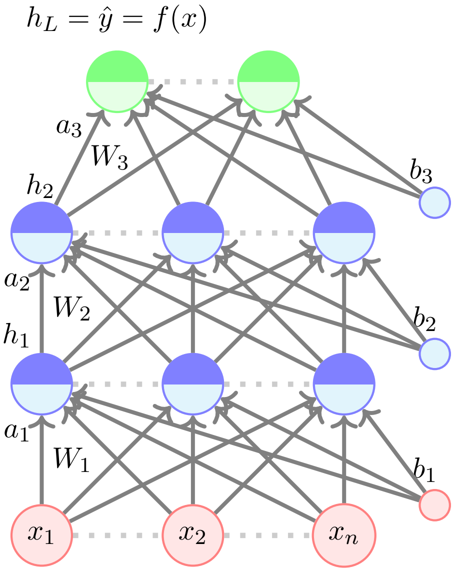



- Notice that $y$ is a probability distribution
- Therefore we should also ensure that $\hat{y}$ is a probability distribution
- What choice of the output activation '$O$' will ensure this ?

$$a_L = W_L h_{L-1} + b_L$$

$$\hat{y}_j = O(a_L)_j = \frac{e^{a_{L,j}}}{\sum_{i=1}^{k} e^{a_{L,i}}}$$

$O(a_L)_j$ is the $j^{th}$ element of $\hat{y}$ and $a_{L,j}$ is the $j^{th}$ element of the vector $a_L$.



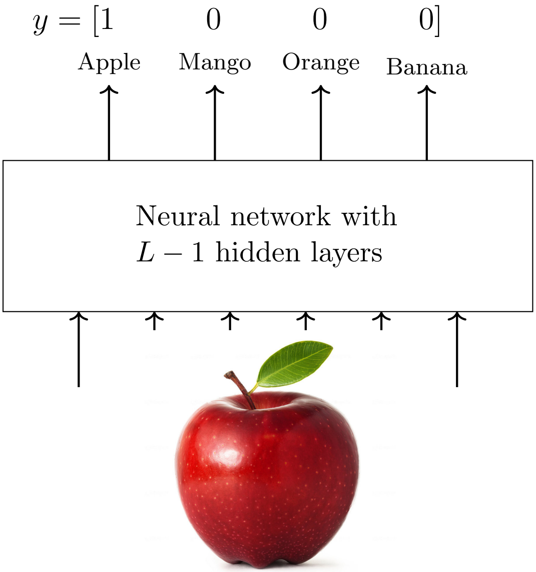

- Notice that $y$ is a probability distribution
- Therefore we should also ensure that $\hat{y}$ is a probability distribution
- What choice of the output activation '$O$' will ensure this ?

$$a_L = W_L h_{L-1} + b_L$$

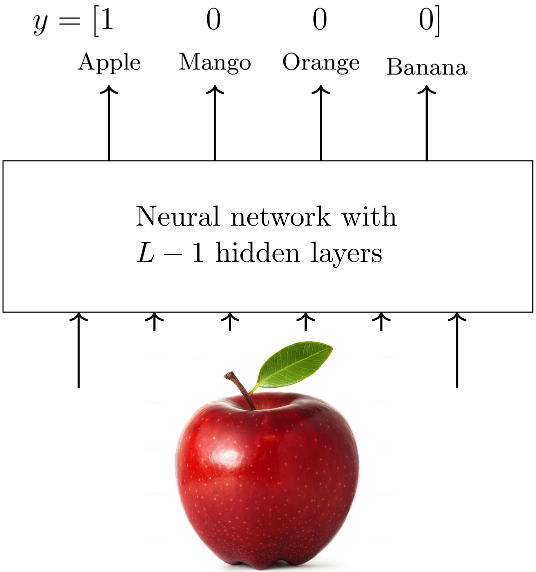
$$\hat{y}_j = O(a_L)_j = \frac{e^{a_{L,j}}}{\sum_{i=1}^{k} e^{a_{L,i}}}$$

$O(a_L)_j$ is the $j^{th}$ element of $\hat{y}$ and $a_{L,j}$ is the $j^{th}$ element of the vector $a_L$.

- This function is called the *softmax* function

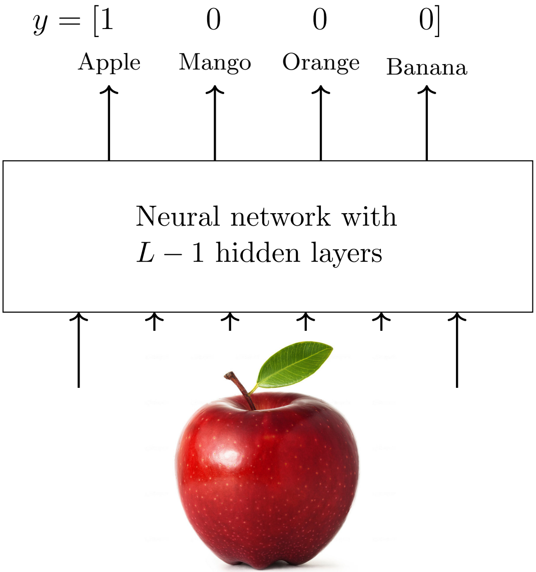



- Now that we have ensured that both $y$ & $\hat{y}$ are probability distributions can you think of a function which captures the difference between them?




- Now that we have ensured that both $y$ & $\hat{y}$ are probability distributions can you think of a function which captures the difference between them?
- Cross-entropy

$$\mathscr{L}(\theta) = -\sum_{c=1}^{k} y_c \log \hat{y}_c$$



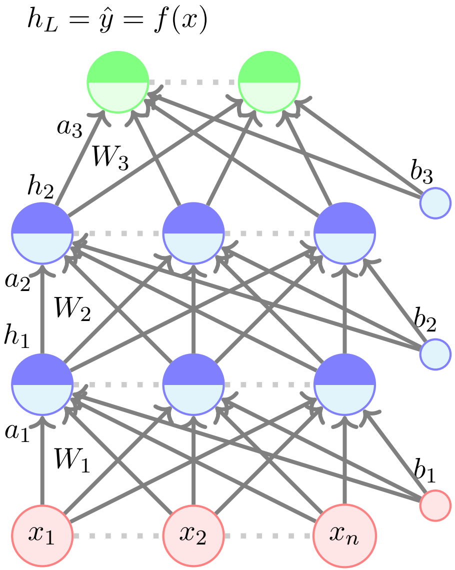

- Now that we have ensured that both $y$ & $\hat{y}$ are probability distributions can you think of a function which captures the difference between them?
- Cross-entropy

$$\mathscr{L}(\theta) = -\sum_{c=1}^{k} y_c \log \hat{y}_c$$

- Notice that

$$\begin{aligned} y_c &= 1 \quad \text{if } c = \ell \text{ (the true class label)} \\ &= 0 \quad \text{otherwise} \end{aligned}$$

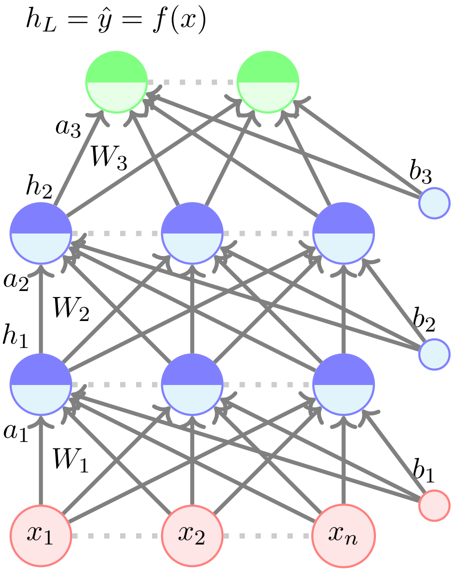
$$\therefore \quad \mathscr{L}(\theta) = -\log \hat{y}_\ell$$




- So, for classification problem (where you have to choose 1 of $K$ classes), we use the following objective function

$$\underset{\theta}{\text{minimize}} \quad \mathscr{L}(\theta) = -\log \hat{y}_\ell$$

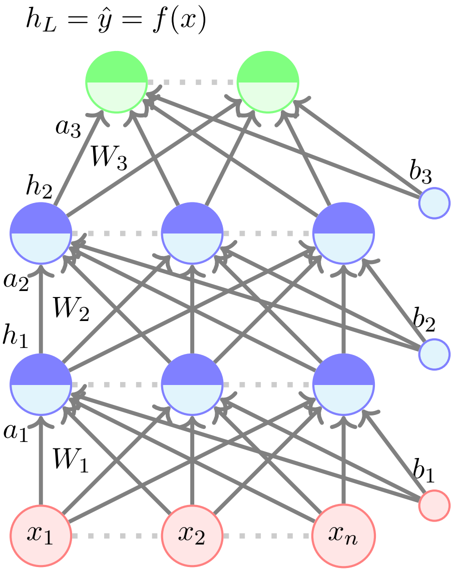
$$\text{or} \quad \underset{\theta}{\text{maximize}} \quad -\mathscr{L}(\theta) = \log \hat{y}_\ell$$




- So, for classification problem (where you have to choose 1 of $K$ classes), we use the following objective function

$$\begin{array}{rcl} & \underset{\theta}{\text{minimize}} & \mathscr{L}(\theta) = -\log \hat{y}_\ell \\ \text{or} & \underset{\theta}{\text{maximize}} & -\mathscr{L}(\theta) = \log \hat{y}_\ell \end{array}$$

- But wait!
  Is $\hat{y}_\ell$ a function of $\theta = [W_1, W_2, .., W_L, b_1, b_2, .., b_L]$?



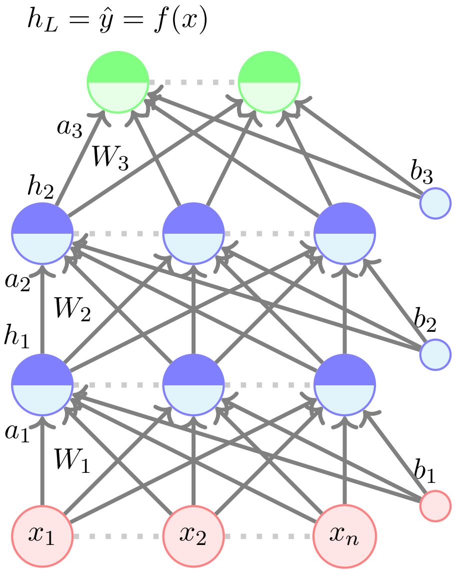

- So, for classification problem (where you have to choose 1 of $K$ classes), we use the following objective function

$$\underset{\theta}{\text{minimize}} \quad \mathscr{L}(\theta) = -\log \hat{y}_\ell$$

$$\text{or} \quad \underset{\theta}{\text{maximize}} \quad -\mathscr{L}(\theta) = \log \hat{y}_\ell$$

- But wait!
  Is $\hat{y}_\ell$ a function of $\theta = [W_1, W_2, ., W_L, b_1, b_2, ., b_L]$?
- Yes, it is indeed a function of $\theta$

$$\hat{y}_\ell = [O(W^3 g(W^2 g(W^1 x + b_1) + b_2) + b_3)]_\ell$$




- So, for classification problem (where you have to choose 1 of $K$ classes), we use the following objective function

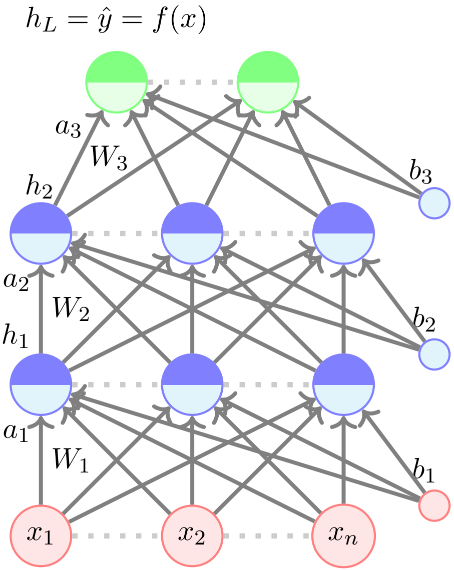
$$\underset{\theta}{\text{minimize}} \quad \mathscr{L}(\theta) = -\log \hat{y}_\ell$$

$$\text{or} \quad \underset{\theta}{\text{maximize}} \quad -\mathscr{L}(\theta) = \log \hat{y}_\ell$$

- But wait!
  Is $\hat{y}_\ell$ a function of $\theta = [W_1, W_2, .., W_L, b_1, b_2, .., b_L]$?
- Yes, it is indeed a function of $\theta$

$$\hat{y}_\ell = [O(W^3 g(W^2 g(W^1 x + b_1) + b_2) + b_3)]_\ell$$

- What does $\hat{y}_\ell$ encode?



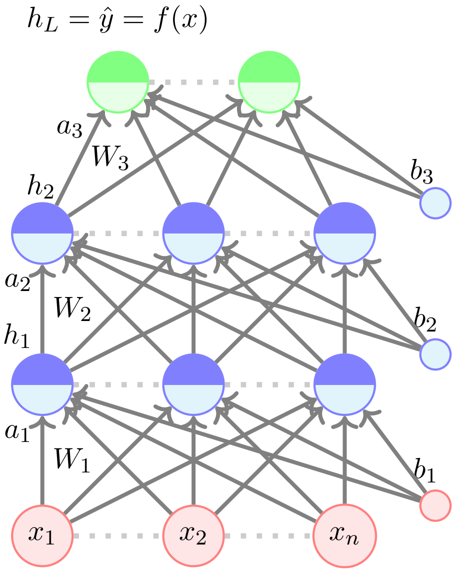

- So, for classification problem (where you have to choose 1 of $K$ classes), we use the following objective function

$$\begin{array}{lll} & \underset{\theta}{\text{minimize}} & \mathscr{L}(\theta) = -\log \hat{y}_\ell \\ \text{or} & \underset{\theta}{\text{maximize}} & -\mathscr{L}(\theta) = \log \hat{y}_\ell \end{array}$$

- But wait!
  Is $\hat{y}_\ell$ a function of $\theta = [W_1, W_2, .., W_L, b_1, b_2, .., b_L]$?
- Yes, it is indeed a function of $\theta$

$$\hat{y}_\ell = [O(W^3 g(W^2 g(W^1 x + b_1) + b_2) + b_3)]_\ell$$

- What does $\hat{y}_\ell$ encode?
- It is the probability that $x$ belongs to the $\ell^{th}$ class (bring it as close to 1).




- So, for classification problem (where you have to choose 1 of $K$ classes), we use the following objective function

$$\begin{aligned} & \underset{\theta}{\text{minimize}} & \mathscr{L}(\theta) = -\log \hat{y}_\ell \\ \text{or} \quad & \underset{\theta}{\text{maximize}} & -\mathscr{L}(\theta) = \log \hat{y}_\ell \end{aligned}$$

- But wait!
  Is $\hat{y}_\ell$ a function of $\theta = [W_1, W_2, ., W_L, b_1, b_2, ., b_L]$?
- Yes, it is indeed a function of $\theta$

$$\hat{y}_\ell = [O(W^3 g(W^2 g(W^1 x + b_1) + b_2) + b_3)]_\ell$$

- What does $\hat{y}_\ell$ encode?
- It is the probability that $x$ belongs to the $\ell^{th}$ class (bring it as close to 1).
- $\log \hat{y}_\ell$ is called the *log-likelihood* of the data.

| | **Outputs** | |
|---|---|---|
| | Real Values | Probabilities |
| Output Activation | | |
| Loss Function | | |

| | Outputs | |
|---|---|---|
| | Real Values | Probabilities |
| Output Activation | Linear | |
| Loss Function | | |

| | **Outputs** | |
|---|---|---|
| | Real Values | Probabilities |
| Output Activation | Linear | Softmax |
| Loss Function | | |

| | Outputs | |
|---|---|---|
| | Real Values | Probabilities |
| Output Activation | Linear | Softmax |
| Loss Function | Squared Error | |

| | **Outputs** | |
|---|---|---|
| | Real Values | Probabilities |
| Output Activation | Linear | Softmax |
| Loss Function | Squared Error | Cross Entropy |

| | Outputs | |
|---|---|---|
| | Real Values | Probabilities |
| Output Activation | Linear | Softmax |
| Loss Function | Squared Error | Cross Entropy |

- Of course, there could be other loss functions depending on the problem at hand but the two loss functions that we just saw are encountered very often

| | Outputs | |
|---|---|---|
| | Real Values | Probabilities |
| Output Activation | Linear | Softmax |
| Loss Function | Squared Error | Cross Entropy |

- Of course, there could be other loss functions depending on the problem at hand but the two loss functions that we just saw are encountered very often
- For the rest of this lecture we will focus on the case where the output activation is a softmax function and the loss function is cross entropy