Configuration of Neural Networks

J.H. Lee, Dept. of Software, Sungkyunkwan Univ.

Contents

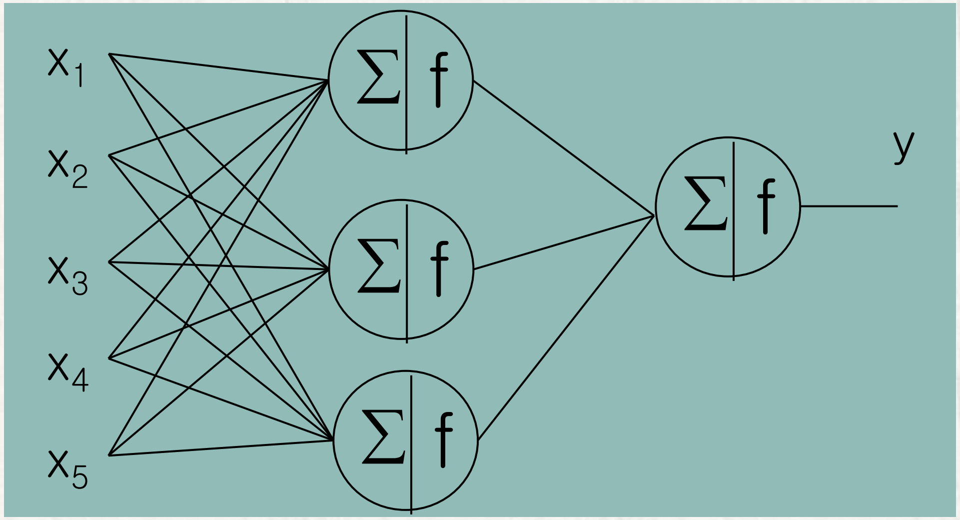
Regression

- Regression: Stock Index Prediction

(2500,2550, 2530, 2540, 2550) -> 2600

(2400,2410, 2420, 2430, 2440) -> 2450

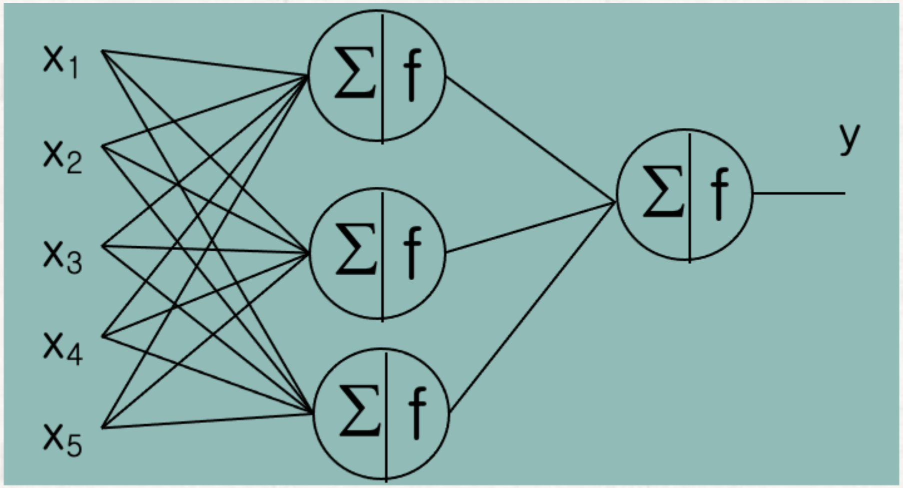
(2470,2460, 2450, 2470, 2480) -> 2470



J.H. Lee, Dept. of Software, Sungkyunkwan Univ.

Regression

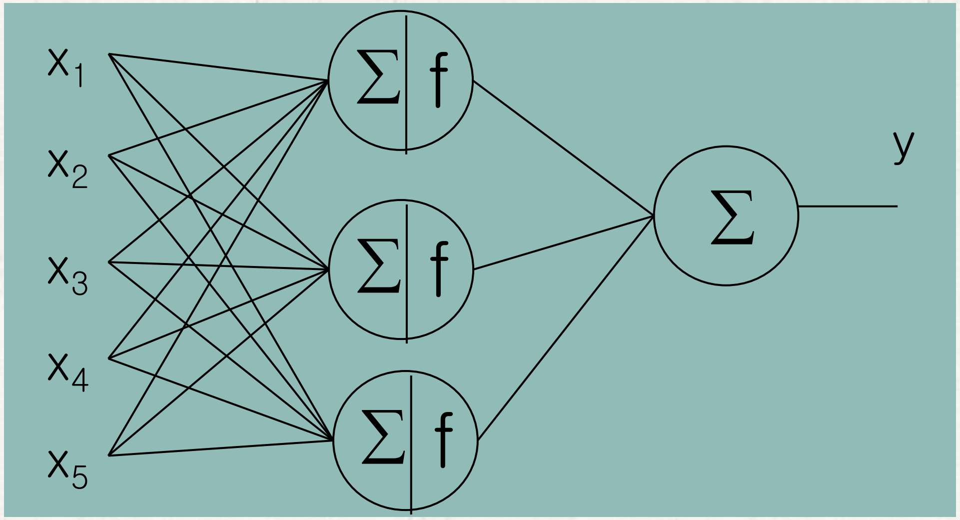
- Following Neural Network is OK for regression?



  - Maybe NO!! Why?
  - The activation functions produces a value between [0,1]

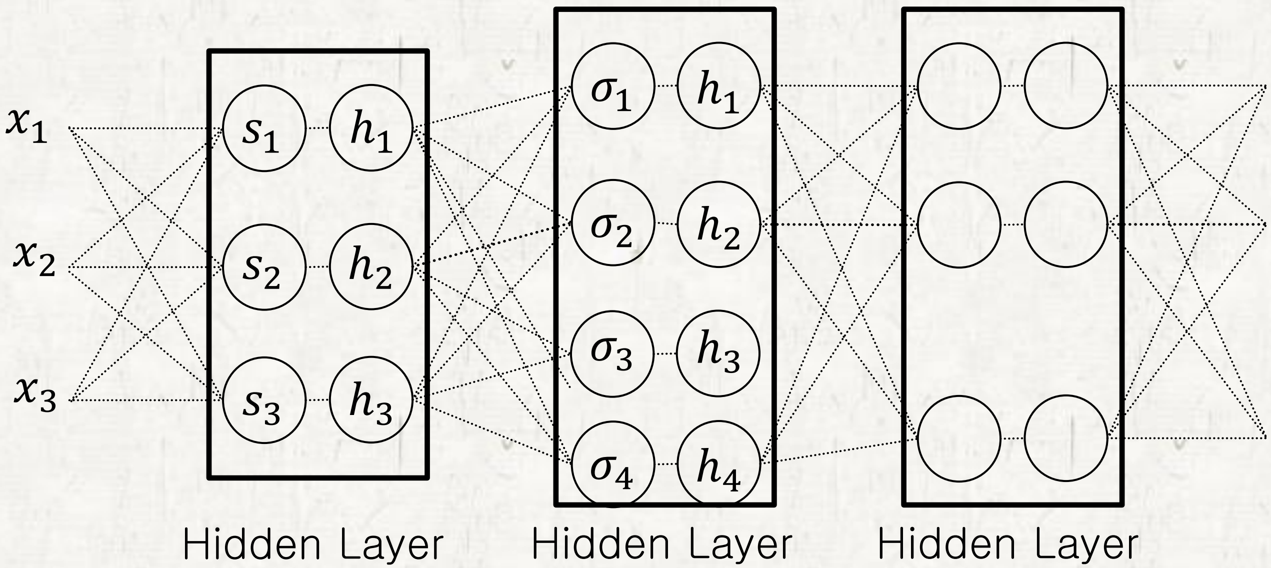
Regression

- Solution
  - Use a linear output node



Regression

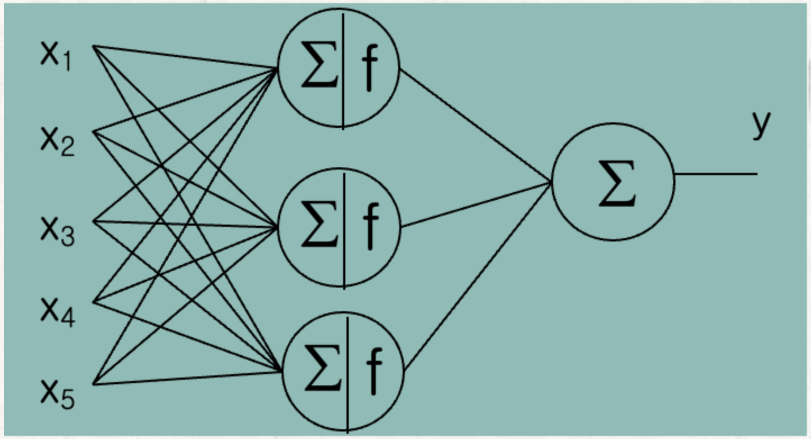
- What layers do



Regression

- Summary

① Linear Activation Function at output node



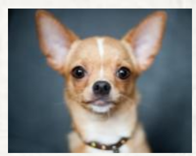
② MSE Loss Function

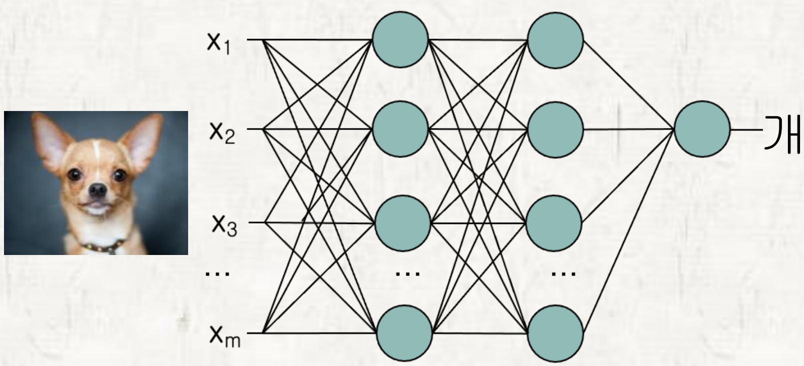
$$E = \sum_{n=1}^{N} (t_n - y_n)^2$$

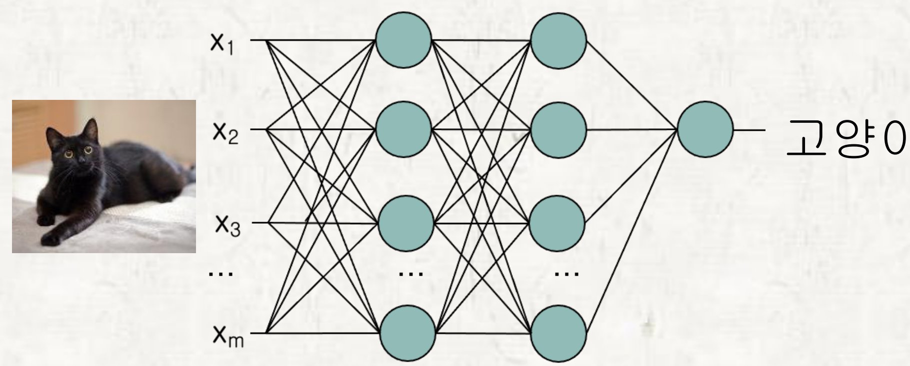
J.H. Lee, Dept. of Software, Sungkyunkwan Univ.

Binary-Class Classification

- Binary-Class Classification:
  - Choosing one from two classes

( , 개) ( , 고양이)

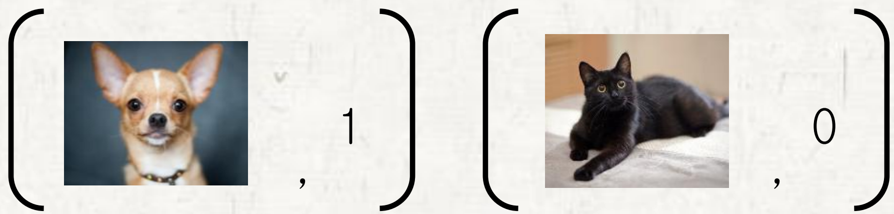


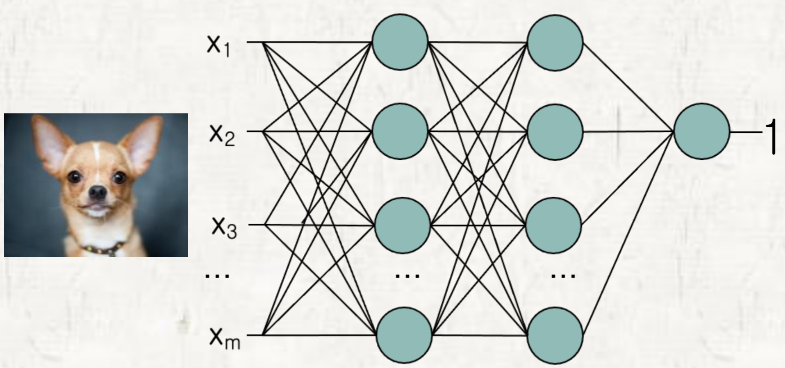


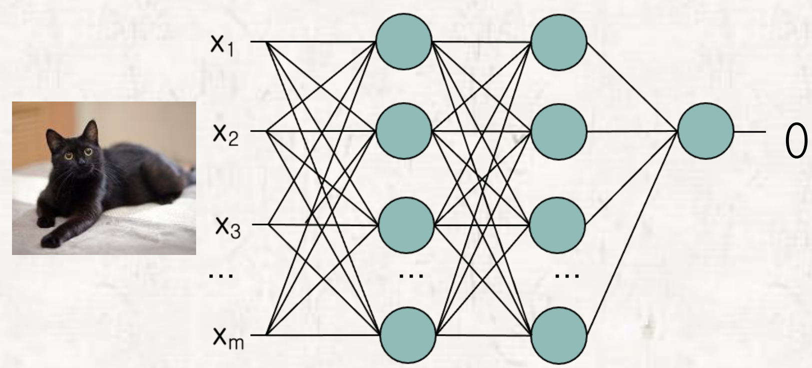
J.H. Lee, Dept. of Software, Sungkyunkwan Univ.

Binary-Class Classification

- Binary-Class Classification:
  - Choosing one from two classes





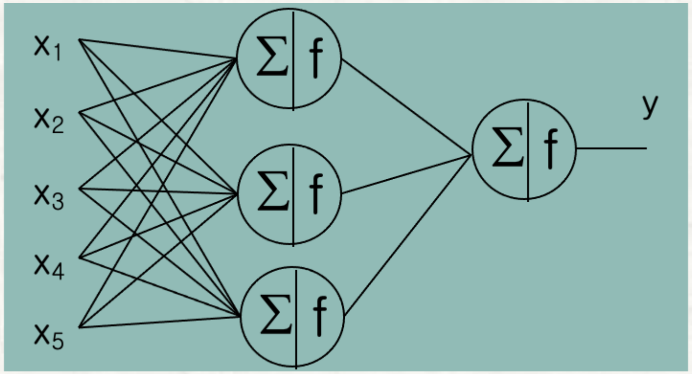


Binary-Class Classification

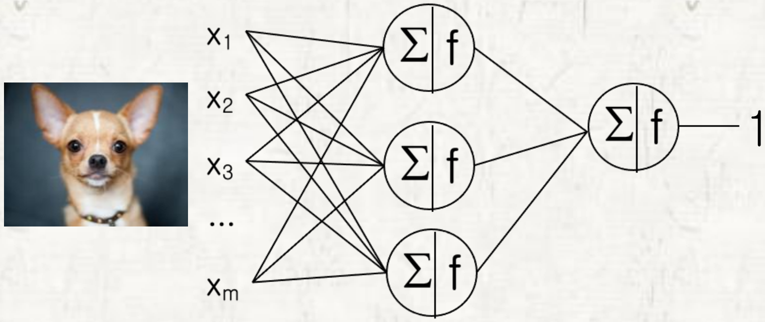
- Categorical Value -> 1 or 0 로 변환

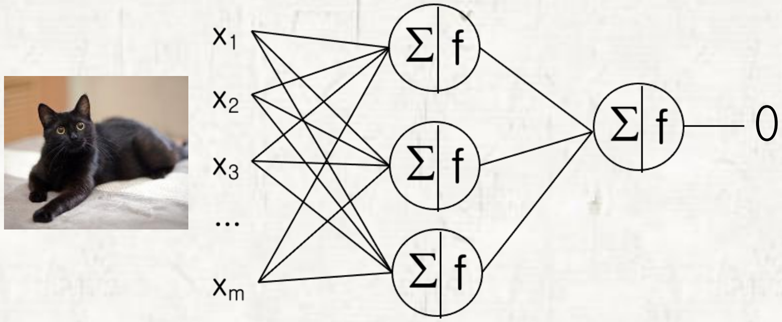
Binary-Class Classification

- Activation Function of output nodes



  - Sigmoid is OK, but it means probability

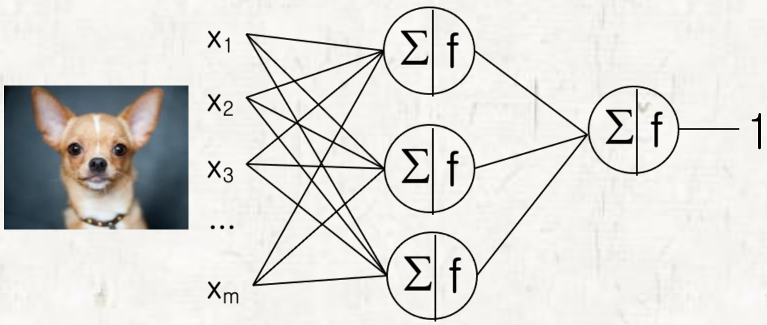


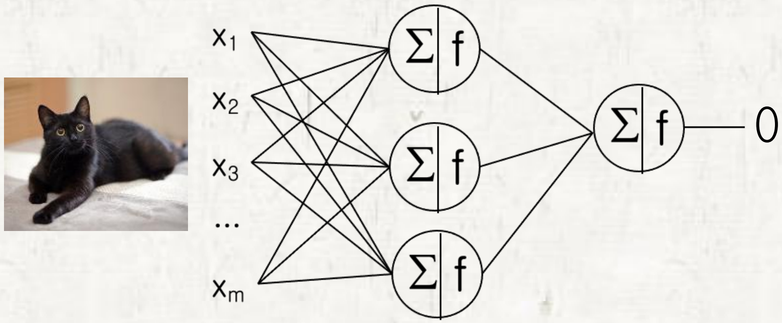


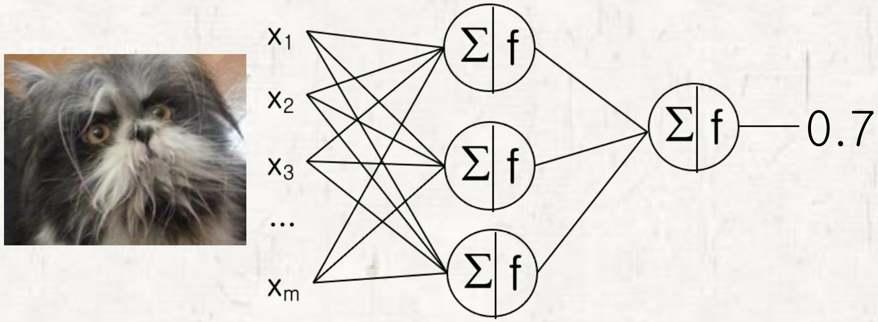
J.H. Lee, Dept. of Software, Sungkyunkwan Univ.

Binary-Class Classification

- Activation Function of output nodes





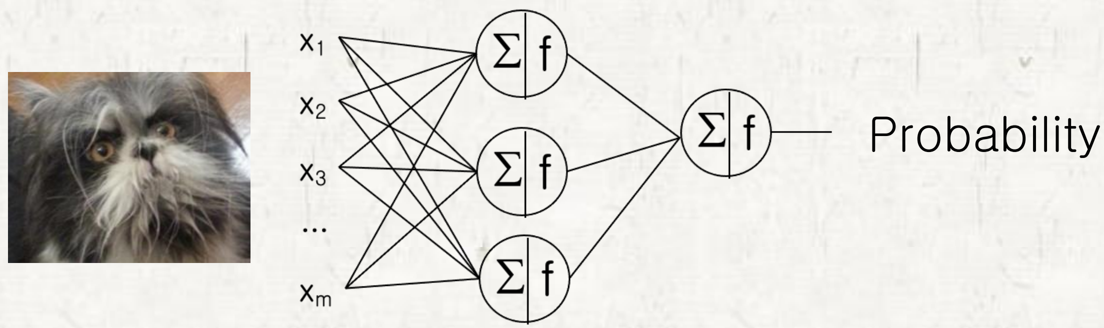


The output is regarded as “Probability of Dog”

J.H. Lee, Dept. of Software, Sungkyunkwan Univ.

Binary-Class Classification

- Loss Function
  - The output means probability



  - Cross Entropy

$$E = -\sum_{n=1}^{N}\left(t_n \log(y_n) + (1 - t_n)\log(1 - y_n)\right)$$

where $t_n \in \{0,1\}$ and $y_n \in [0,1]$

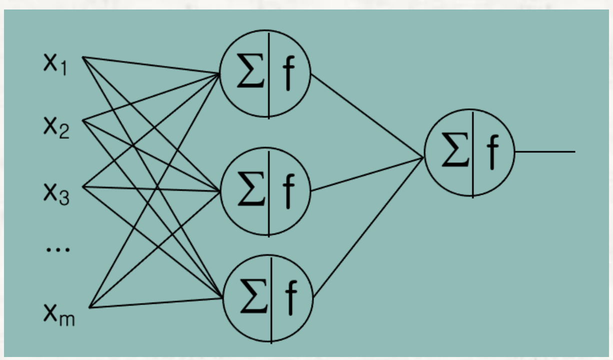
Binary-Class Classification

- Summary

① Preprocessing

( , 개) ( , 고양이) → ( , 1) ( , 0)

② Sigmoid at output node



③ Cross Entropy

$$E = -\sum_{n=1}^{N} \left(t_n \log(y_n) + (1 - t_n) \log(1 - y_n) \right)$$

where $t_n \in \{0,1\}$ and $y_n \in [0,1]$

Multi-Class Classification

- We have 4 classes

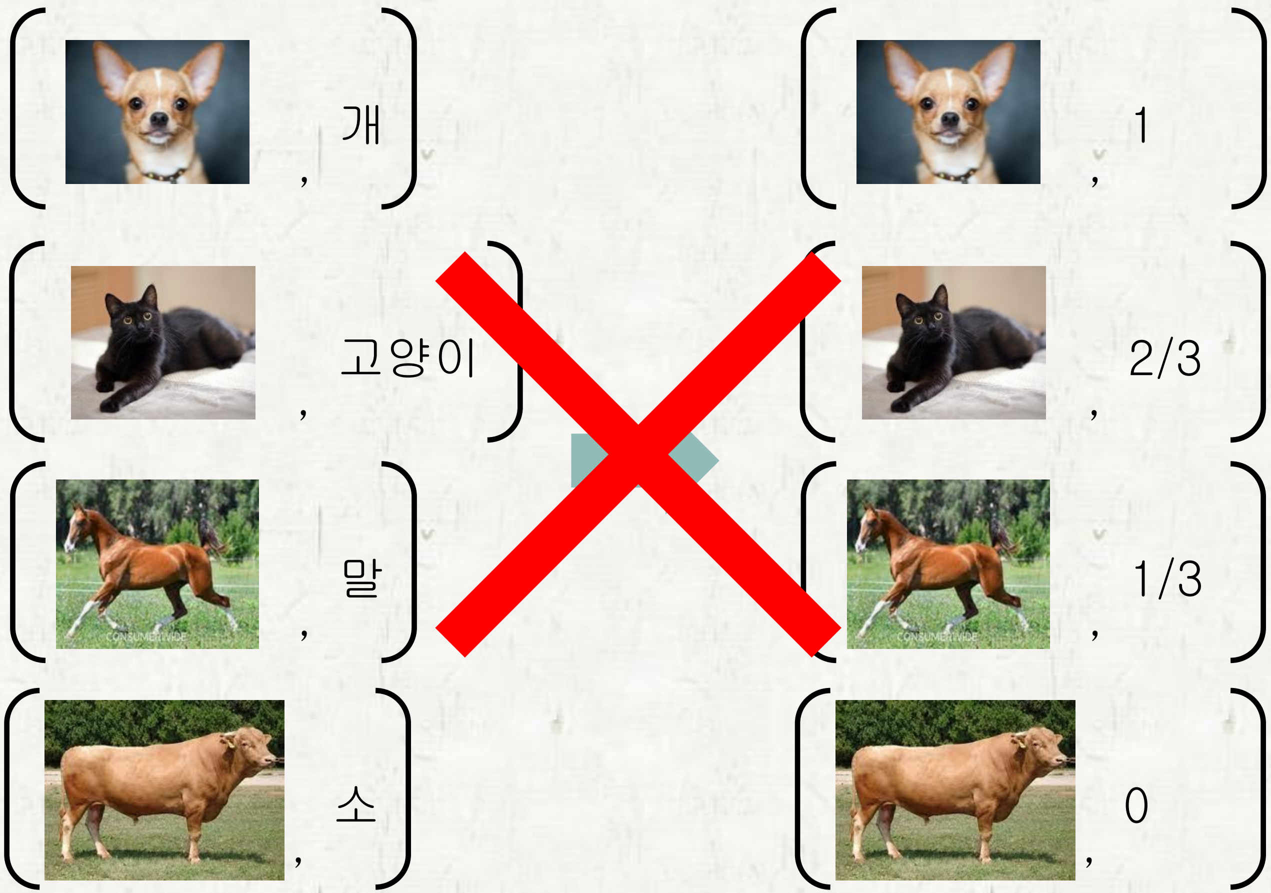
( , 개) ( , 고양이)

( , 말) ( , 소)

- How to Conver Categorical Values

Multi-Class Classification

- Categorical Value Conversion



J.H. Lee, Dept. of Software, Sungkyunkwan Univ.

Multi-Class Classification

- Not Good… why?

개 → 1.0

고양이 → 2/3

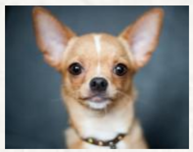
말 → 1/3

소 → 0.0

J.H. Lee, Dept. of Software, Sungkyunkwan Univ.

Multi-Class Classification

- Categorical Value Conversion: One-hot-encoding

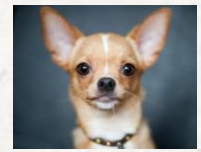
( , 개)

( , 고양이)

( , 말)

( , 소)

→

( , 1, 0, 0, 0)

( , 0, 1, 0, 0)

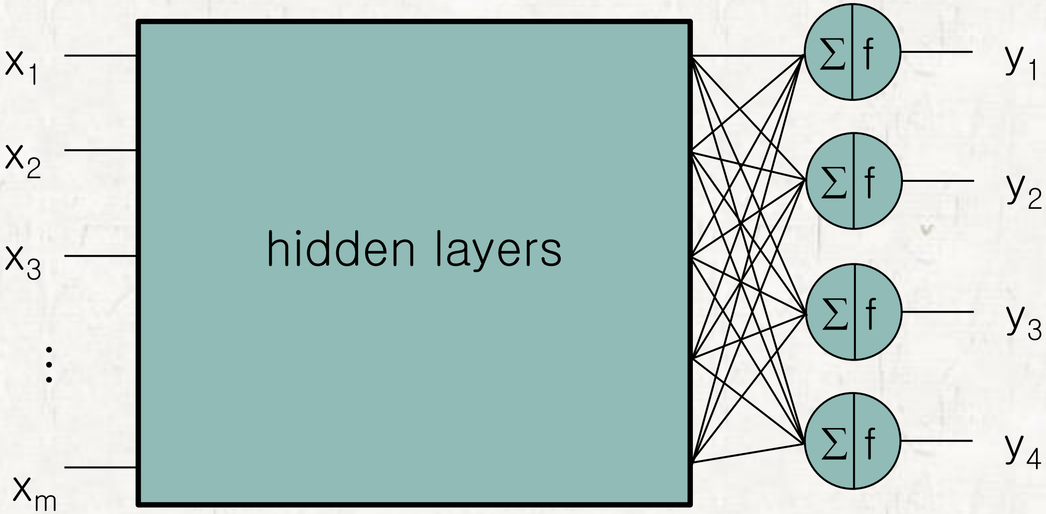
( , 0, 0, 1, 0)

( , 0, 0, 0, 1)

Multi-Class Classification

- Structure of Neural Network

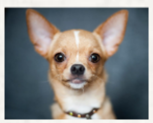
( , 1, 0, 0, 0) ( , 0, 1, 0, 0) ( , 0, 0, 1, 0) ( , 0, 0, 0, 1)

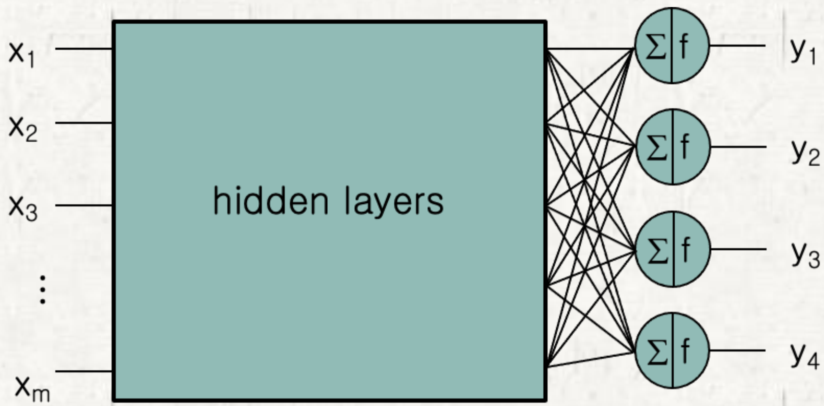


Multi-Class Classification

- Activation Function of Output nodes

f = Softmax





- Loss Function: Cross Entropy

$$E = \sum_{n=1}^{Data} \sum_{k=1}^{Class} -t_{nk} \log(y_{nk})$$

Multi-Class Classification

③ Choose the softmax instead of max

$$(y_1, y_2 , y_3) = \mathrm{softmax}(x_1, x_2, x_3)$$

$$y_k = \frac{e^{x_k}}{\sum_{i=1}^{n} e^{x_i}}$$

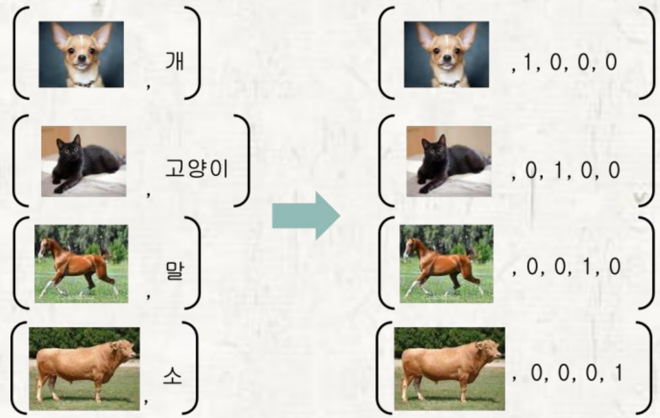
y_1	y_2	y_3
0.301	0.332	0.367
0.090	0.245	0.665
0.042	0.114	0.844
0.017	0.047	0.936
0.000	0.000	1.000
0.000	0.000	1.000

x_1	x_2	x_3
1	1.1	1.2
1	2	3
1	2	4
1	2	5
1	2	10
1	2	20

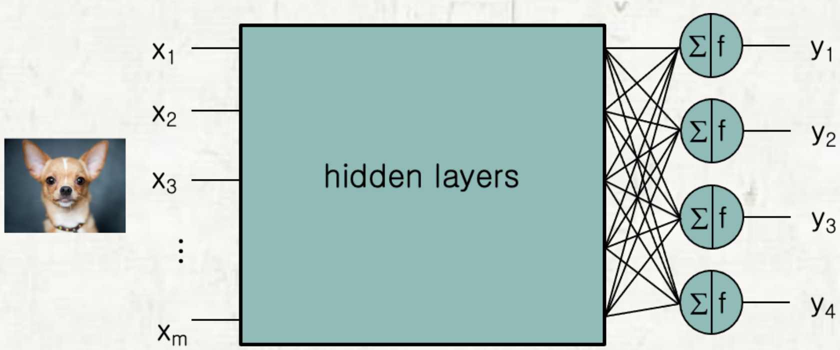
Multi-Class Classification

- Summary

① One-hot-encoding



② Use softmax



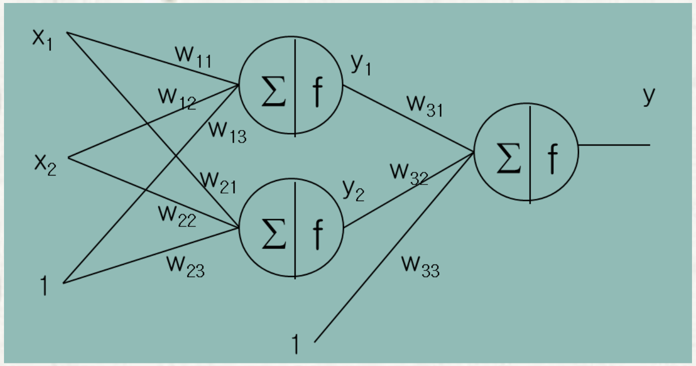
③ Use cross entropy

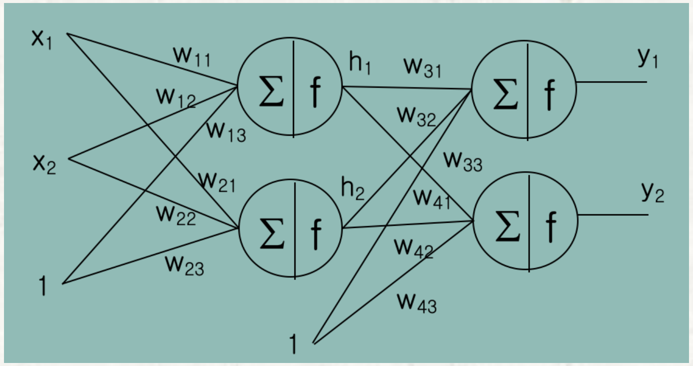
$$E = \sum_{n=1}^{Data} \sum_{k=1}^{Class} -t_{nk} \log(y_{nk})$$

Multi-Class Classification

- Cross Entropy for Multi-Class

(x_{11}, x_{12}, Red)	$(x_{11}, x_{12}, 1)$	$(x_{11}, x_{12}, 1, 0)$
(x_{21}, x_{22}, Red)	$(x_{21}, x_{22}, 1)$	$(x_{21}, x_{22}, 1, 0)$
$(x_{31}, x_{32}, Black)$	$(x_{31}, x_{32}, 0)$	$(x_{31}, x_{32}, 0, 1)$
(x_{41}, x_{42}, Red)	$(x_{41}, x_{42}, 1)$	$(x_{41}, x_{42}, 1, 0)$
$(x_{51}, x_{52}, Black)$	$(x_{51}, x_{52}, 0)$	$(x_{51}, x_{52}, 0, 1)$





$$-(t_n \log(y_n) + (1 - t_n)\log(1 - y_n))$$

$$-(t_{n1}\log(y_{n1}) + t_{n2}\log(y_{n2})) = -\sum_{k=1}^{Class} t_{nk}\log(y_{nk})$$

J.H. Lee, Dept. of Software, Sungkyunkwan Univ.

Nominal Inputs

- What if you have categorical inputs
  - Two inputs and one output

$$x_1 \in R$$
$$x_2 \in \{Red, Yellow, Blue\}$$
$$y \in \{0,1\}$$

  - Create a new input variable for each categorical value

$$x_2 = \begin{cases} 1 & \text{if original } x_2 \text{ is Yellow} \\ 0 & \text{Otherwise} \end{cases}$$

$$x_3 = \begin{cases} 1 & \text{if original } x_2 \text{ is Red} \\ 0 & \text{Otherwise} \end{cases}$$

$$x_4 = \begin{cases} 1 & \text{if original } x_2 \text{ is Blue} \\ 0 & \text{Otherwise} \end{cases}$$

$$\begin{array}{l} (0.1, \text{Red}, 0) \\ (0.2, Blue, 1) \\ (0.3, Yellow, 0) \\ (0.4, Red, 1) \end{array} \Rightarrow \begin{array}{l} (0.1, 1,0,0, 0) \\ (0.2, 0,0,1, 1) \\ (0.3, 0,1,0, 0) \\ (0.4, 1,0,0, 1) \end{array}$$

Summary

Problem	Activation Function		Loss function
	Hidden Layer	Output Layer	
Regression	ReLU	Linear	MSE
2-class Classification	ReLU	Sigmoid	CE
Multi-class Classification	ReLU	Softmax	CE