# Configuration of Neural Networks



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

# Contents

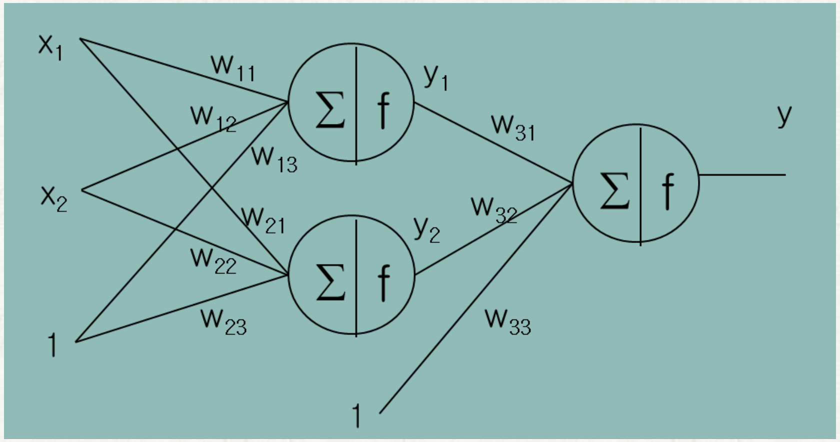




J.H. Lee, Dept. of Software, Sungkyunkwan Univ.

# Regression

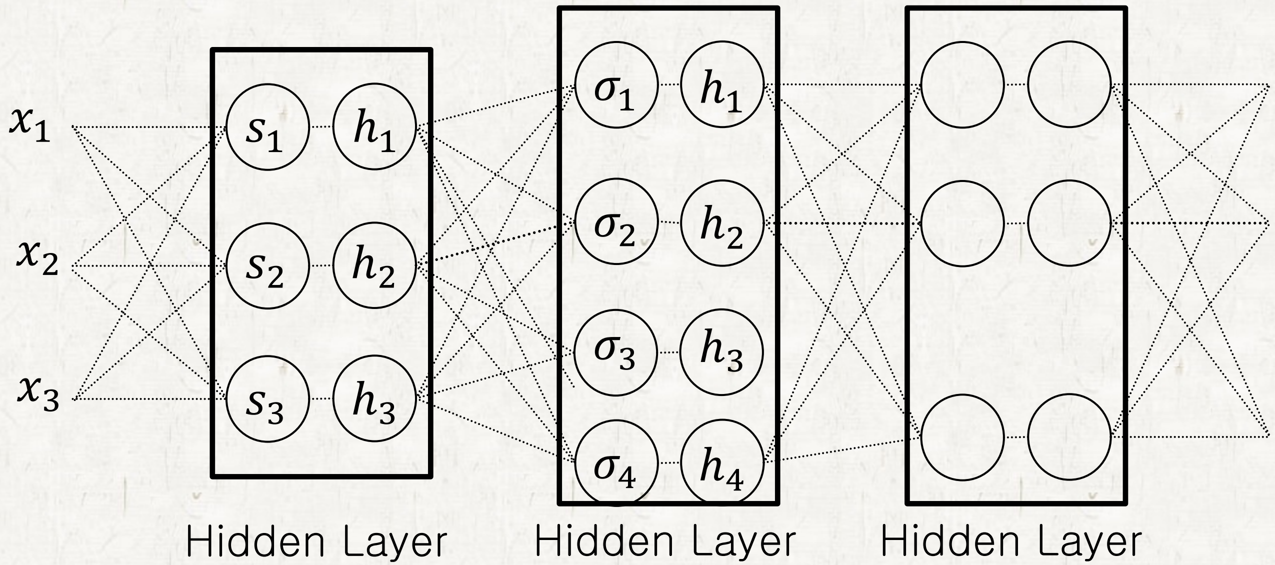
- Following Neural Network is OK for regression?




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

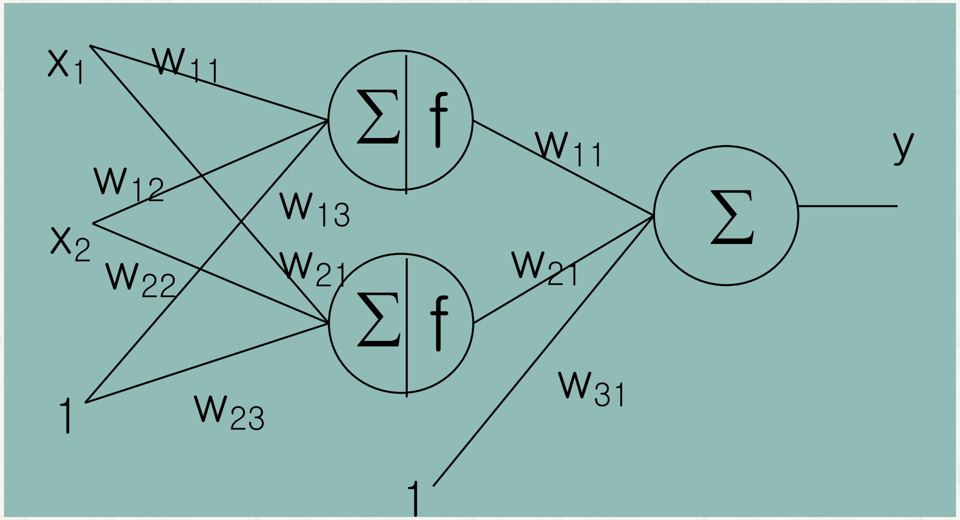
# Regression

- What layers do



# Regression

- Solution
  - Use a linear output node



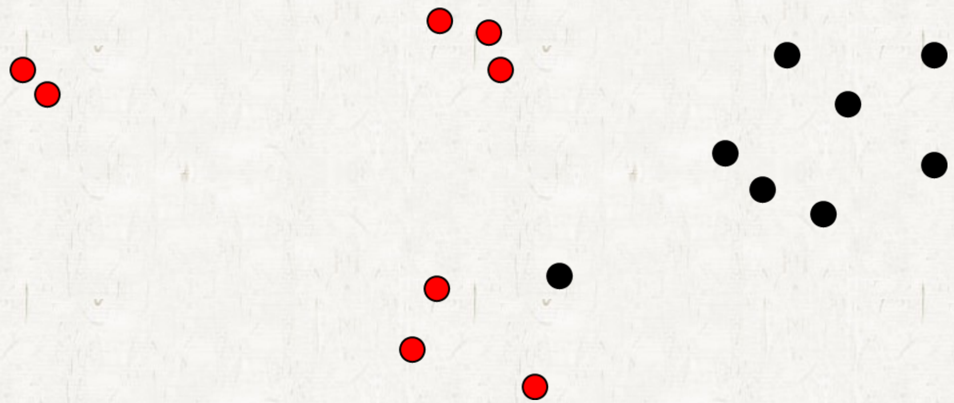



J.H. Lee, Dept. of Software, Sungkyunkwan Univ.

# Binary-Class Classification

- You Have Two Problems

$(\mathrm{x}_{11}, \mathrm{x}_{12}, Red),\ (\mathrm{x}_{21}, \mathrm{x}_{22}, Red),\ (\mathrm{x}_{31}, \mathrm{x}_{32}, Black),\ (\mathrm{x}_{41}, \mathrm{x}_{42}, Red), (\mathrm{x}_{51}, \mathrm{x}_{52}, Black), \cdots$



  - P1: NN cannot produces nominal values
  - P2: Error Function for training

# Binary-Class Classification

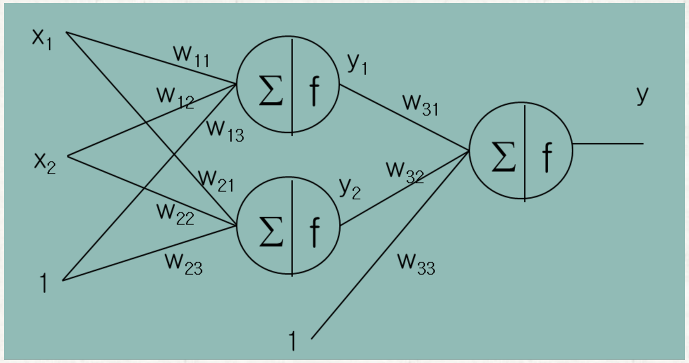
- P1: Handling Nominal Values
  - Use 0 and 1 for class labels

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

$$\Downarrow$$

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

  - Use Sigmoid





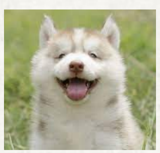

J.H. Lee, Dept. of Software, Sungkyunkwan Univ.

# Binary-Class Classification

- But.. There is a problem

Training Data

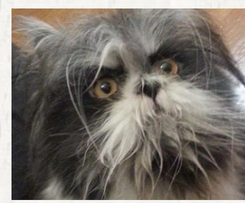
( , 0 )



( , 1 )

...

Test Data



NN → 0.7

I call this "Label", then How can we interpret this?



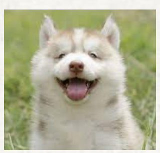
J.H. Lee, Dept. of Software, Sungkyunkwan Univ.

# Binary-Class Classification

- But.. There is a problem

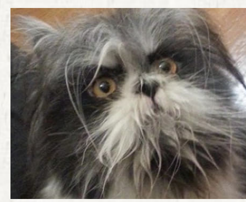
Training Data

(  , 0 )

(  , 1 )

...

Test Data



NN

0.7

Let's regard this as "Probability of Dog", then it is easy to interpret

# Binary-Class Classification

- But.. There is another problem !
  - Output of Classification -> Probability
    - We need to describe the training process using “probablility”

Find w so that NN correctly predicts all training data

⇔ Find w which maximizes the probability that NN correctly predicts all training data

⇔ Find w which maximizes the following:

$$\left(\prod_{(x,1)\in Data} NN(\mathrm{x};\mathrm{w})\right) \times \left(\prod_{(x,0)\in Data} (1 - NN(\mathrm{x};\mathrm{w}))\right)$$

# Binary-Class Classification

- Cross Entropy

$$\underset{\mathrm{w}}{\operatorname{argmax}}\left(\prod_{(x,1)\in Data} NN(\mathrm{x};\mathrm{w}) \times \prod_{(x,0)\in Data} (1 - NN(\mathrm{x};\mathrm{w}))\right)$$

$$= \underset{\mathrm{w}}{\operatorname{argmax}} \log\left(\prod_{(x,1)\in Data} NN(\mathrm{x};\mathrm{w}) \times \prod_{(x,0)\in Data} (1 - NN(\mathrm{x};\mathrm{w}))\right)$$

$$= \underset{\mathrm{w}}{\operatorname{argmax}}\left(\sum_{(x,1)\in Data} \log NN(\mathrm{x};\mathrm{w}) + \sum_{(x,0)\in Data} \log(1 - NN(\mathrm{x};\mathrm{w}))\right)$$

$$= \underset{\mathrm{w}}{\operatorname{argmax}}\left(\sum_{(x,1)\in Data} y \log NN(\mathrm{x};\mathrm{w}) + (1-y)\log(1 - NN(\mathrm{x};\mathrm{w})) + \sum_{(x,0)\in Data} y \log NN(\mathrm{x};\mathrm{w}) + (1-y)\log(1 - NN(\mathrm{x};\mathrm{w}))\right)$$

$$= \underset{\mathrm{w}}{\operatorname{argmax}}\left(\sum_{(x,y)\in Data} y \log NN(\mathrm{x};\mathrm{w}) + (1-y)\log(1 - NN(\mathrm{x};\mathrm{w}))\right)$$

$$= \underset{\mathrm{w}}{\operatorname{argmin}}\left(-\sum_{(x,y)\in Data} y \log NN(\mathrm{x};\mathrm{w}) + (1-y)\log(1 - NN(\mathrm{x};\mathrm{w}))\right)$$

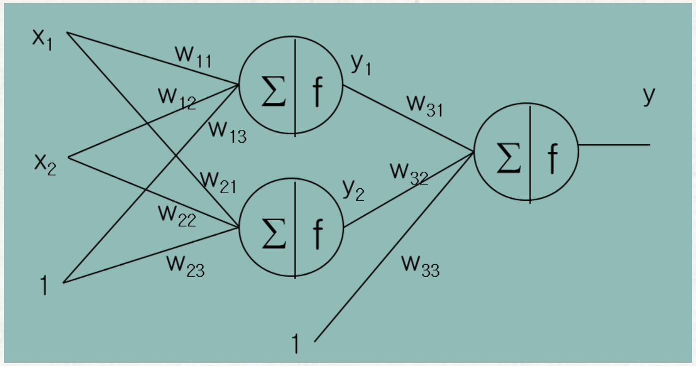
# Binary-Class Classification

- Summary

① Preprocessing

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

② Sigmoid at output node




③ Cross Entropy

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

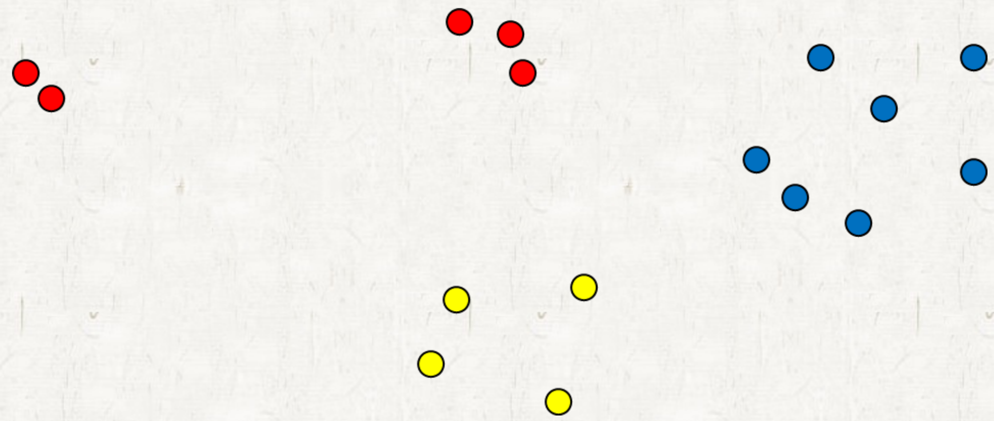
where $t_n \in \{0,1\}$ and $y_n \in [0,1]$


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

# Multi-Class Classification

- Problem

$$(\mathbf{x}_1, Red),\ (\mathbf{x}_2, Yellow),\ (\mathbf{x}_3, Blue),\ (\mathbf{x}_4, Red), (\mathbf{x}_5, Blue), \cdots$$



- Easy…



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

# Multi-Class Classification

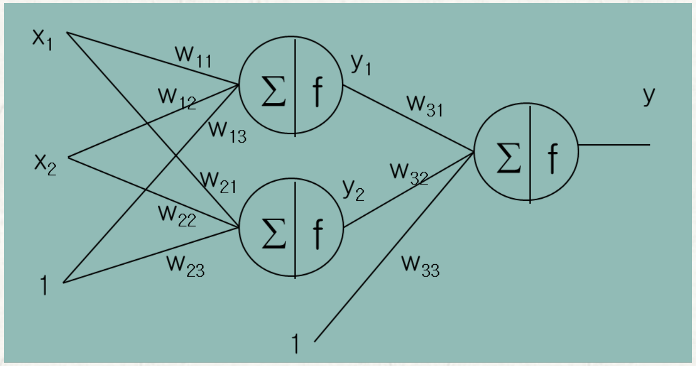
- Nominal Value Handling: Linear conversion of class labels

$(x_{11}, x_{12}, Red)$, $(x_{21}, x_{22}, Yellow)$, $(x_{31}, x_{32}, Blue)$, $(x_{41}, x_{42}, Red)$, $(x_{51}, x_{52}, Blue)$, $\cdots$

$\Downarrow$

$(x_{11}, x_{12}, 1)$, $(x_{21}, x_{22}, 0.5)$, $(x_{31}, x_{32}, 0)$, $(x_{41}, x_{42}, 1)$, $(x_{51}, x_{52}, 0)$, $\cdots$

- Use Sigmoid at output node




- Prediction

$$class(\mathbf{x}) = \begin{cases} 1 & NN(\mathbf{x}) \geq 2/3 \\ 0.5 & 2/3 \geq NN(\mathbf{x}) \geq 1/3 \\ 0 & Otherwise \end{cases}$$



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

# Multi-Class Classification

- Not Good… why?
  - There is no order between Red, Yellow, Blue
  - They are just names. We cannot say that Red > Yellow > Blue
  - Linear conversion changes the original problem.

$$\text{Red} \longrightarrow 1.0$$

$$\text{Yellow} \longrightarrow 0.5$$

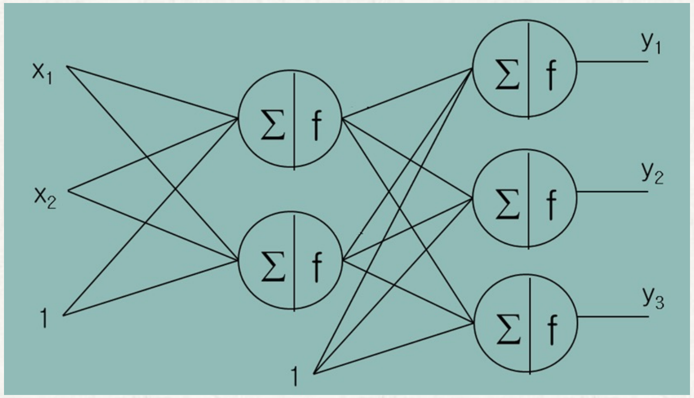
$$\text{Blue} \longrightarrow 0.0$$

# Multi-Class Classification

- Then?
  - ① Create virtual outputs

$$\begin{array}{lcl} (x_{11}, x_{12}, Red), & & (x_{11}, x_{12}, 1, 0, 0), \\ (x_{21}, x_{22}, Yellow), & & (x_{21}, x_{22}, 0, 1, 0), \\ (x_{31}, x_{32}, Blue), & \Rightarrow & (x_{31}, x_{32}, 0, 0, 1), \\ (x_{41}, x_{42}, Red), & & (x_{41}, x_{42}, 1, 0, 0), \\ (x_{51}, x_{52}, Blue), & & (x_{51}, x_{52}, 0, 0, 1), \\ \ldots & & \ldots \end{array}$$

  - ② Place nodes at the output layer as many as outputs

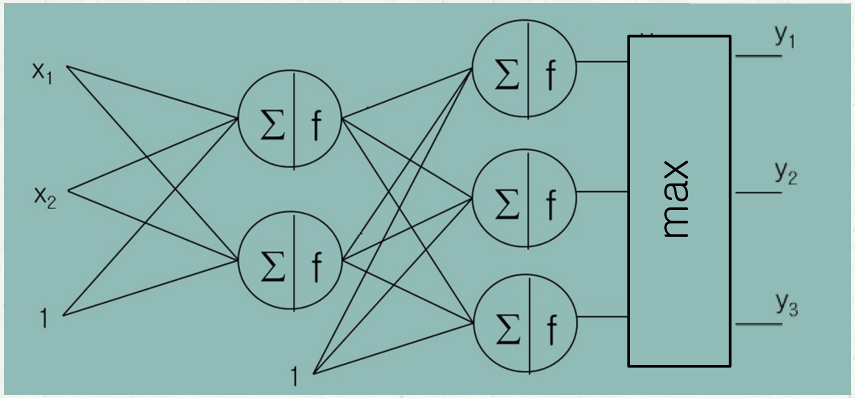





J.H. Lee, Dept. of Software, Sungkyunkwan Univ.

# Multi-Class Classification

③ Choose the maximum



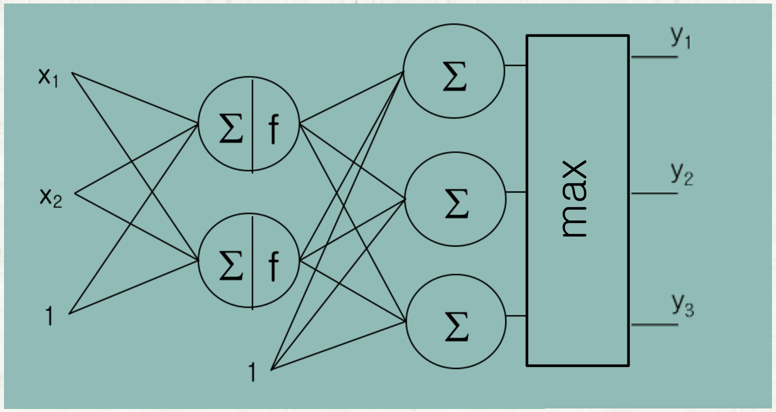

Redefine “max” to return the position of the maximum value

$$(1, 0, 0) = \max(10, 5, 2)$$

# Multi-Class Classification

③ Choose the maximum (con'd)

Since the sigmoid is monotonically increasing, we have the same output if we remove the sigmoid activations.




But.. "max" is not differentiable.

Let's find other function similar to "max" but differentiable.



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

# Multi-Class Classification

③ Choose the softmax instead of max

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

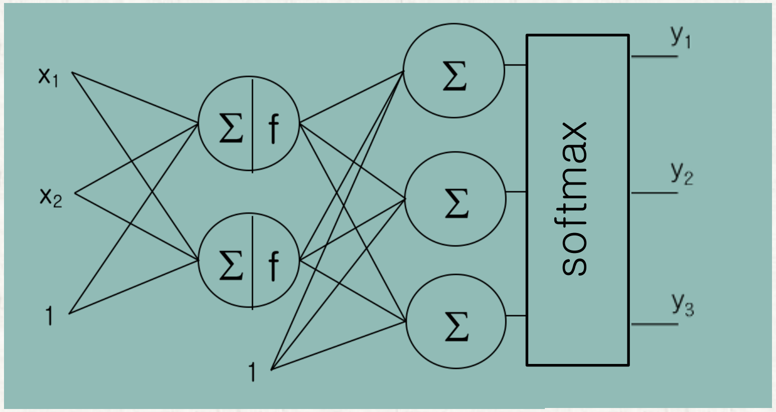
$$y_k = \frac{e^{x_k}}{\sum_{i=1}^{n} e^{x_i}}$$

| $y_1$ | $y_2$ | $y_3$ |
|---|---|---|
| 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

- Loss Function




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

Hmm?? $-\left(t_n \log(y_n) + (1 - t_n)\log(1 - y_n)\right)$

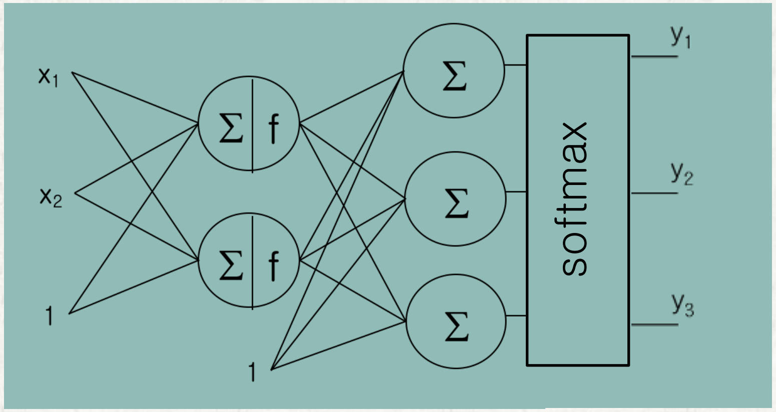
# Multi-Class Classification

- Summary

① Create virtual outputs

$$\begin{array}{l}(x_{11}, x_{12}, Red),\\(x_{21}, x_{22}, Yellow),\\(x_{31}, x_{32}, Blue),\\(x_{41}, x_{42}, Red),\\(x_{51}, x_{52}, Blue),\\\ldots\end{array} \Rightarrow \begin{array}{l}(x_{11}, x_{12}, 1, 0, 0),\\(x_{21}, x_{22}, 0, 1, 0),\\(x_{31}, x_{32}, 0, 0, 1),\\(x_{41}, x_{42}, 1, 0, 0),\\(x_{51}, x_{52}, 0, 0, 1),\\\ldots\end{array}$$

② Use softmax




③ Use cross entropy

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

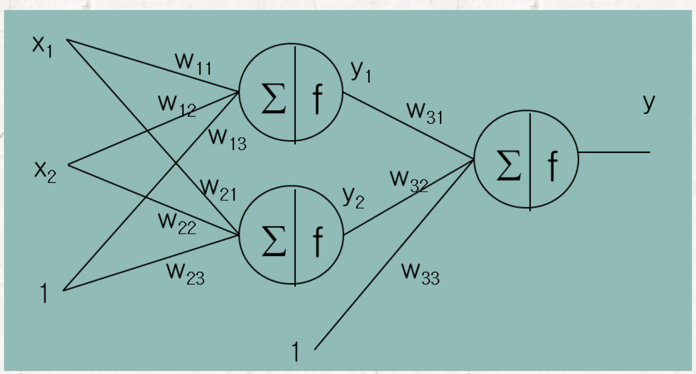


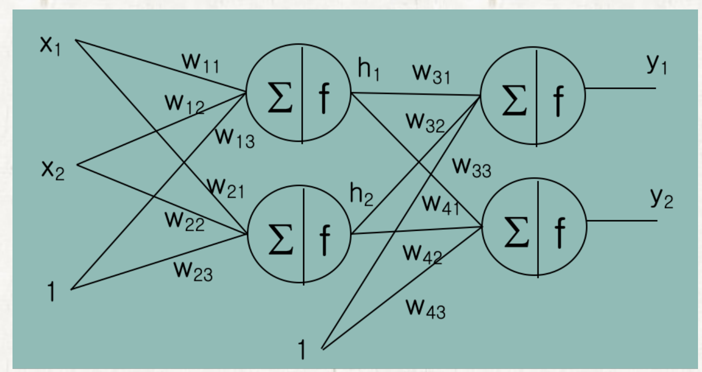
J.H. Lee, Dept. of Software, Sungkyunkwan Univ.

# Multi-Class Classification

- Cross Entropy for Multi-Class

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




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

# 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

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

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

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

$(0.1, \text{Red}, 0)$
$(0.2, Blue, 1)$
$(0.3, Yellow, 0)$
$(0.4, Red, 1)$

$(0.1, 1,0,0, 0)$
$(0.2, 0,0,1, 1)$
$(0.3, 0,1,0, 0)$
$(0.4, 1,0,0, 1)$

# 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 |



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