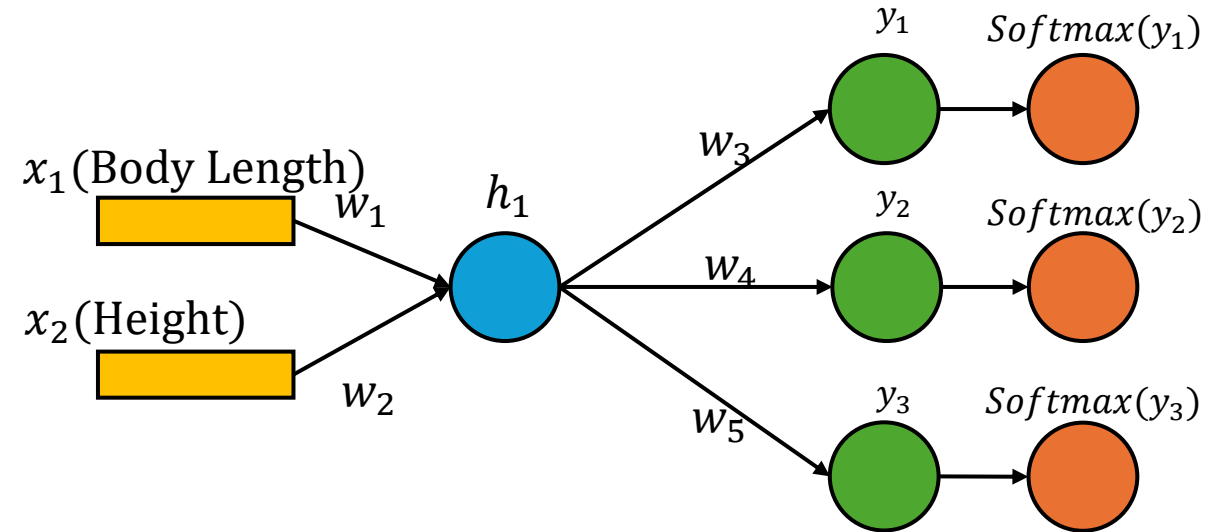


Softmax Cross Entropy



$$CE = - \sum_{i=1}^n \textit{Observed} \cdot \log(P_i)$$

Classification Neural Network



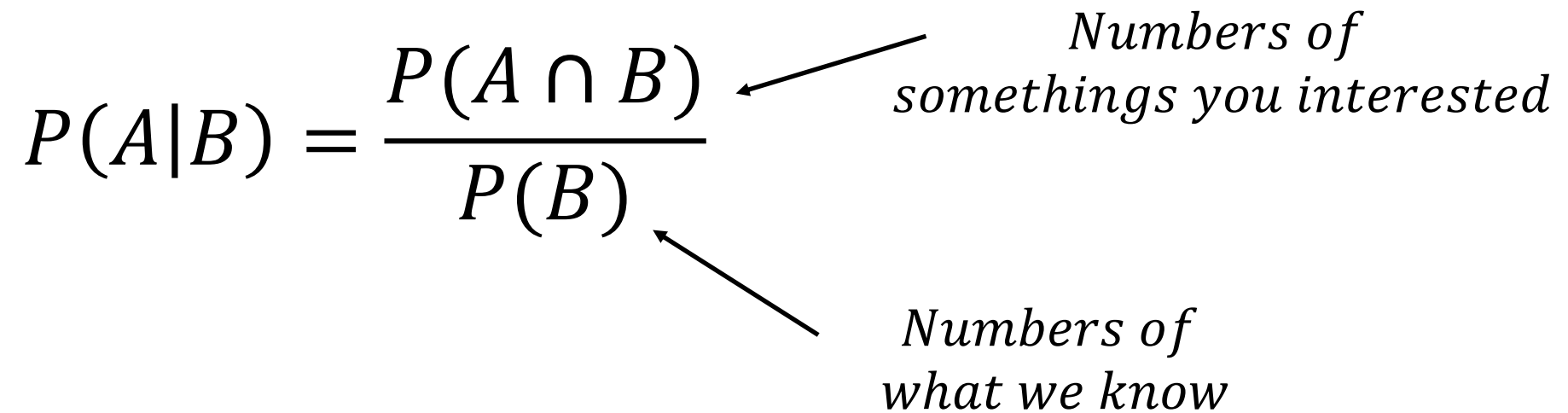
**Why Softmax + Cross Entropy
is works with classification???**

Conditional Probabilities

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

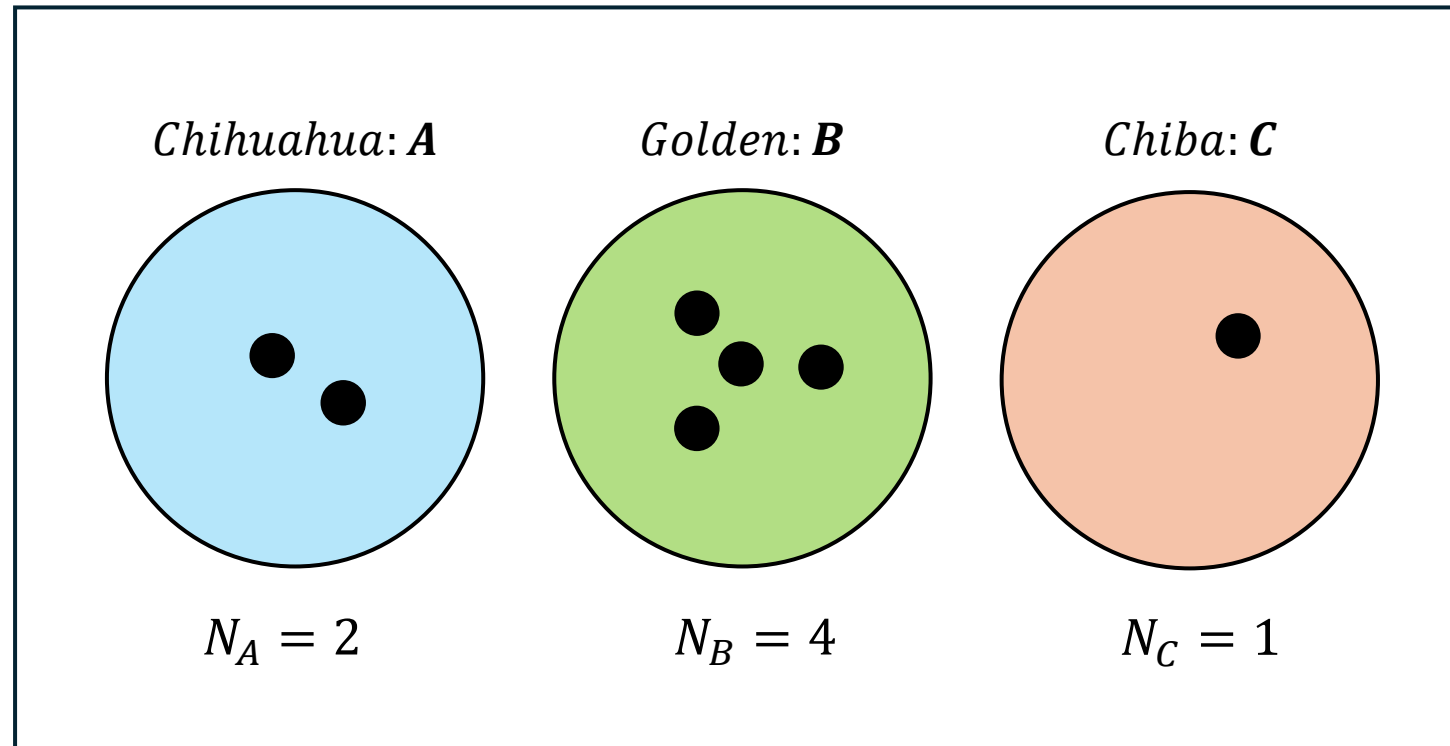
*Numbers of
somethings you interested*

*Numbers of
what we know*

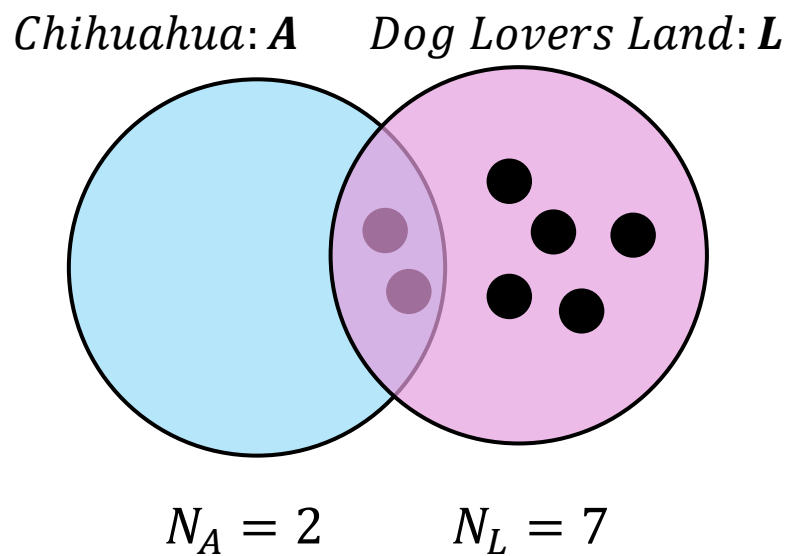
A diagram illustrating the formula for conditional probability. The formula is $P(A|B) = \frac{P(A \cap B)}{P(B)}$. An arrow points from the text "Numbers of somethings you interested" to the numerator $P(A \cap B)$. Another arrow points from the text "Numbers of what we know" to the denominator $P(B)$.

Conditional Probabilities

Dog Lovers Land. (Short in L)



$$\text{Total Populations} = N_A + N_B + N_C = 7$$



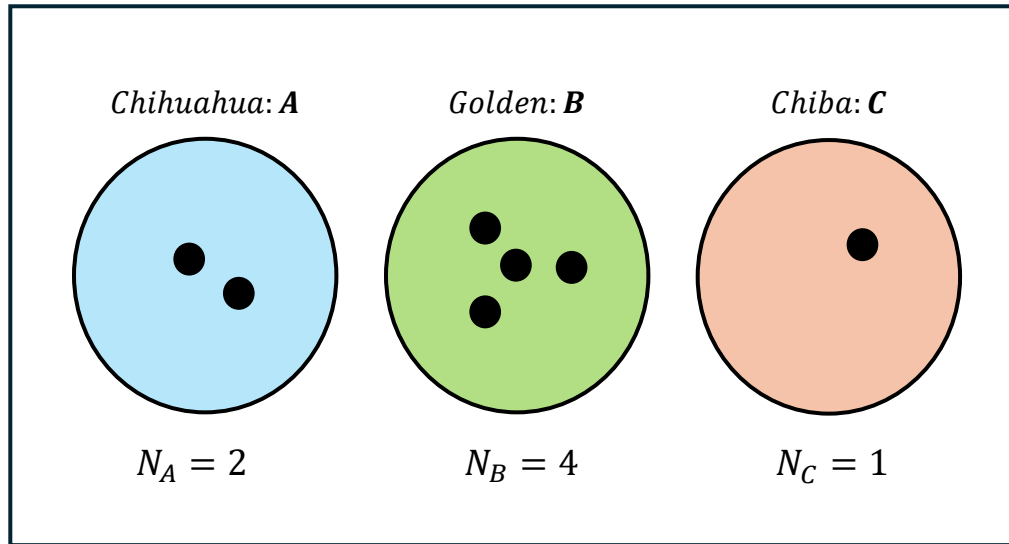
$$P(A|L) = \frac{P(A \cap L)}{P(L)}$$

$$P(A|L) = \frac{\text{Number of } A \cap L}{\text{Number of } L}$$

$$P(A|L) = \frac{2}{7} = 0.29$$

Range of probability: $[0, 1]$

Dog Lovers Land. (Short in L)



Total Populations = $N_A + N_B + N_C = 7$

$$\text{Chihuahua} \rightarrow P(A|L) = \frac{2}{7} = 0.29$$

$$\text{Golden} \rightarrow P(B|L) = \frac{4}{7} = 0.57$$

$$\text{Chiba} \rightarrow P(C|L) = \frac{1}{7} = 0.14$$

$$P(A|L) + P(B|L) + P(C|L) = 0.29 + 0.14 + 0.57$$

$$P(A|L) + P(B|L) + P(C|L) = 1$$

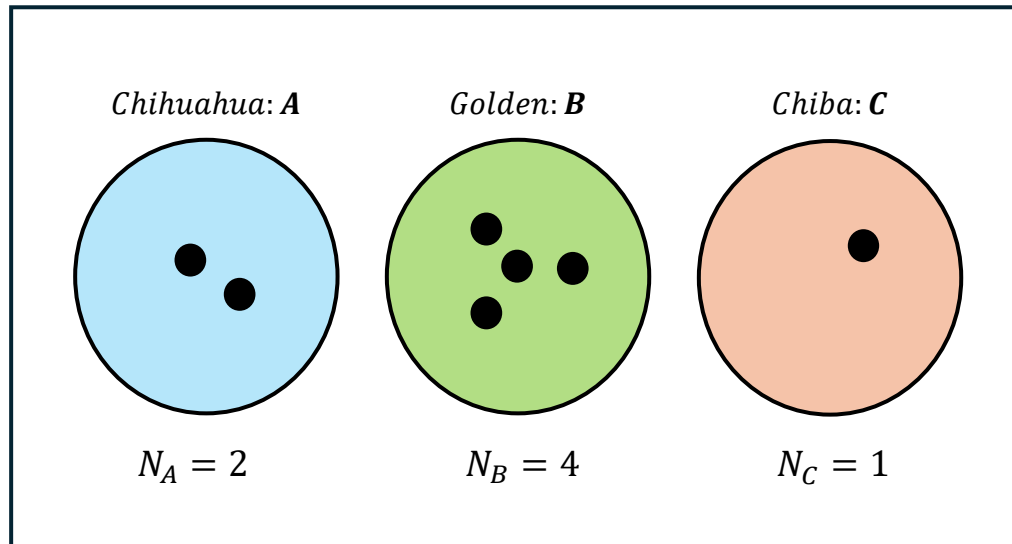
Why we need Softmax ???

Softmax

$$\textit{Softmax}(x_i) = \frac{e^{x_i}}{\sum_{j=1}^n e^{x_j}}$$

It can normalize outputs to scale within the range of $[0, 1]$.

*Dog Lovers Land. (Short in **L**)*



Total Populations = $N_A + N_B + N_C = 7$

$$\text{Softmax}(x_i) = \frac{e^{x_i}}{\sum_{j=1}^n e^{x_j}}$$

$$\text{Softmax}(x_A) = \frac{e^{x_A}}{e^{x_A} + e^{x_B} + e^{x_C}}$$

$$\text{Softmax}(x_B) = \frac{e^{x_B}}{e^{x_A} + e^{x_B} + e^{x_C}}$$

$$\text{Softmax}(x_C) = \frac{e^{x_C}}{e^{x_A} + e^{x_B} + e^{x_C}}$$

$$\textit{Softmax}(x_A) = \frac{e^{x_A}}{e^{x_A} + e^{x_B} + e^{x_C}} = \frac{e^{0.29}}{e^{0.29} + e^{0.57} + e^{0.14}} = 0.31$$

$$\textit{Softmax}(x_B) = \frac{e^{x_B}}{e^{x_A} + e^{x_B} + e^{x_C}} = \frac{e^{0.57}}{e^{0.29} + e^{0.57} + e^{0.14}} = 0.42$$

$$\textit{Softmax}(x_C) = \frac{e^{x_C}}{e^{x_A} + e^{x_B} + e^{x_C}} = \frac{e^{0.14}}{e^{0.29} + e^{0.57} + e^{0.14}} = 0.27$$

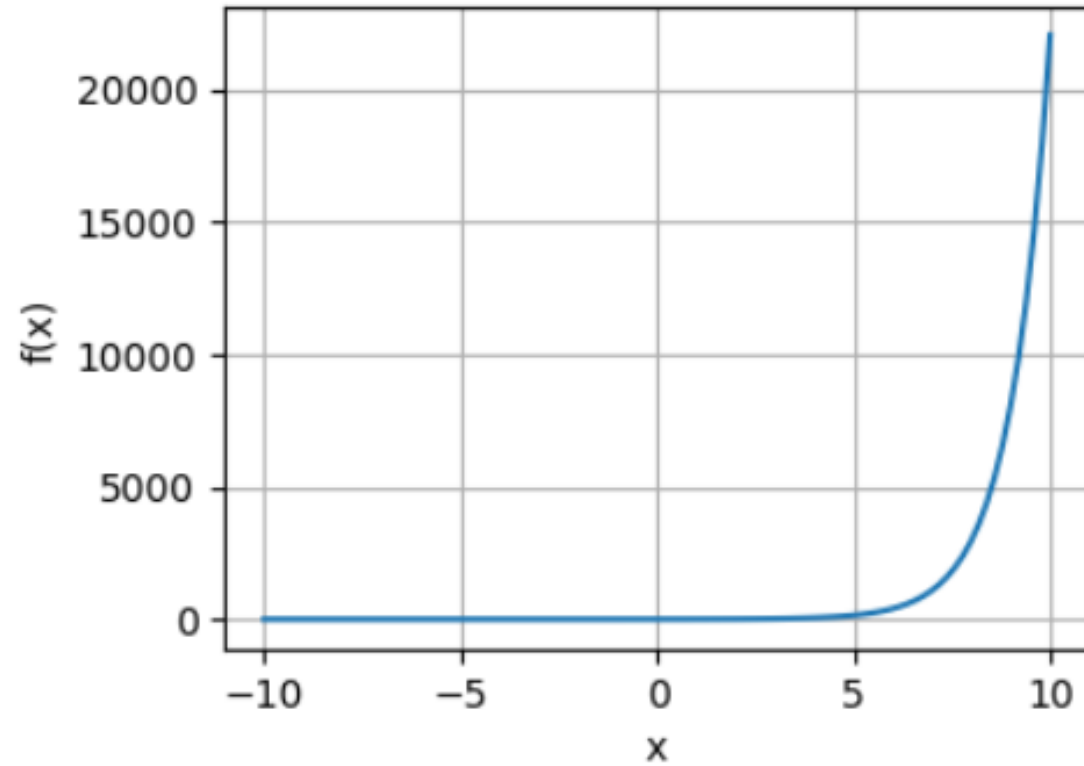
$$\textit{Softmax}(x_A) + \textit{Softmax}(x_B) + \textit{Softmax}(x_C) = 1.0$$

**If you just need to scale it
into range of $[0,1]$**

**Why not we just use
normalization ???**

The output from sum of node
in neural network maybe
negative number

$$f(x) = e^x$$



Range of output always be: $(0, +\infty)$

Furthermore, the **Softmax** can boost the value of output to closer of the expectation.

Example

$$\textit{Output} = [10, 20]$$

$$\textit{Normalization} = \left[\frac{10}{10 + 20}, \frac{20}{10 + 20} \right] = [0.333 \dots, 0.666 \dots]$$

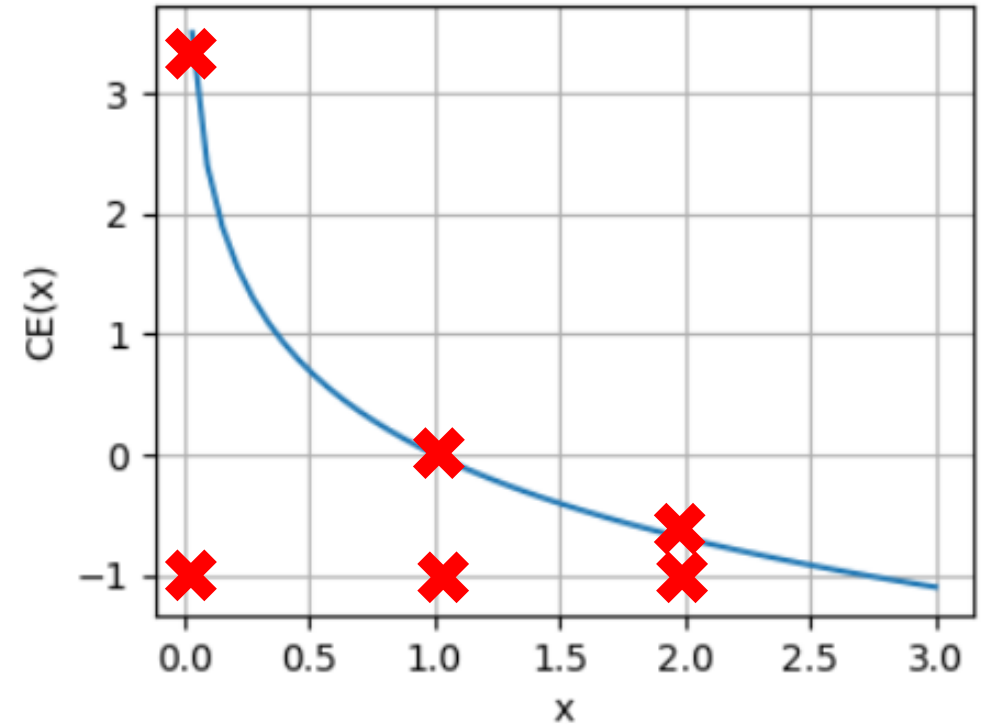
$$\textit{Softmax} = \left[\frac{e^{10}}{e^{10} + e^{20}}, \frac{e^{20}}{e^{10} + e^{20}} \right] = [0.00 \dots, 0.999 \dots]$$

Next, Why we always use
Cross Entropy with
Softmax ???

Cross Entropy

$$CE = - \sum_{i=1}^n Observed \cdot \log(P_i)$$

$$CE = - \sum_{i=1}^n Observed \cdot \log(Softmax_i)$$



Therefore, If we try to minimize
the **Cross Entropy**
The error is going to less.

