

Softmax Regression

Multinomial Logistic Regression (Native Multiclass)

Softmax Function:

$$P(y=k|x) = e^{w_k^T x} / \sum_j e^{w_j^T x}$$



Outputs: **K probabilities** that sum to 1



Generalizes sigmoid to multiple classes



Each class has its own weight vector w_k



More elegant than one-vs-rest for multiclass



Widely used: Neural network output layer

Weight Vectors per Class

w_1 for Class 1

w_2 for Class 2

w_3 for Class 3

... ...

w_K for Class K

Softmax → $\sum P = 1$

vs. Sigmoid

Softmax Regression - Calculation Example

1
2
3
4

3-Class Classification Example

1 Given Input & Weights

입력 벡터: $x = [2, 3]$

Weight vectors:

$$w_1 = [0.5, 0.3] \rightarrow \text{Class 1 (Cat)}$$

$$w_2 = [0.8, 0.4] \rightarrow \text{Class 2 (Dog)}$$

$$w_3 = [0.2, 0.9] \rightarrow \text{Class 3 (Bird)}$$

2 Calculate Logits ($z = w^T x$)

$$z_1 = w_1^T x = 0.5 \times 2 + 0.3 \times 3 = 1.0 + 0.9 = 1.9$$

$$z_2 = w_2^T x = 0.8 \times 2 + 0.4 \times 3 = 1.6 + 1.2 = 2.8$$

$$z_3 = w_3^T x = 0.2 \times 2 + 0.9 \times 3 = 0.4 + 2.7 = 3.1$$

3 Apply Exponential Function

$$e^{z_1} = e^{1.9} \approx 6.686$$

$$e^{z_2} = e^{2.8} \approx 16.445$$

$$e^{z_3} = e^{3.1} \approx 22.198$$

Sum = 6.686 + 16.445 + 22.198 = **45.329**

4 Calculate Softmax Probabilities

$$P(y=1|x) = e^{z_1} / \text{Sum} = 6.686 / 45.329 \approx \textbf{0.147 (14.7\%)}$$

$$P(y=2|x) = e^{z_2} / \text{Sum} = 16.445 / 45.329 \approx \textbf{0.363 (36.3\%)}$$

$$P(y=3|x) = e^{z_3} / \text{Sum} = 22.198 / 45.329 \approx \textbf{0.490 (49.0\%)}$$

Final Result

Prediction: Class 3 (Bird) with 49.0% probability

Note: All probabilities sum to 1.0 ($0.147 + 0.363 + 0.490 = 1.000$)