



Problem 3: In machine learning you often come across problems which contain the following quantity

$$y = \log \sum_{i=1}^N e^{x_i}$$

For example if we want to calculate the log-likelihood of neural network with a softmax output we get this quantity due to the normalization constant. If you try to calculate it naively, you will quickly encounter underflows or overflows, depending on the scale of x_i . Despite working in log-space, the limited precision of computers is not enough and the result will be ∞ or $-\infty$.

To combat this issue we typically use the following identity:

$$y = \log \sum_{i=1}^N e^{x_i} = a + \log \sum_{i=1}^N e^{x_i - a}$$

for an arbitrary a . This means, you can shift the center of the exponential sum. A typical value is setting a to the maximum ($a = \max_i x_i$), which forces the greatest value to be zero and even if the other values would underflow, you get a reasonable result.

Your task is to show that the identity holds.

This is called the log-sum-exp trick and is often used in practice

$$y = \log \sum_{i=1}^N e^{x_i} ;$$

$$e^y = \sum_{i=1}^N e^{x_i} ;$$

$$e^{-a} e^y = e^{-a} \sum_{i=1}^N e^{x_i} ;$$

$$e^{y-a} = \sum_{i=1}^N e^{-a} e^{x_i} ;$$

$$y-a = \log \sum_{i=1}^N e^{x_i - a} ;$$

$$y = a + \log \sum_{i=1}^N e^{x_i - a}$$

Problem 4: Similar to the previous exercise we can compute the output of the softmax function $\pi_i = \frac{e^{x_i}}{\sum_{i=1}^N e^{x_i}}$ in a numerically stable way by shifting by an arbitrary constant a :

$$\frac{e^{x_i}}{\sum_{i=1}^N e^{x_i}} = \frac{e^{x_i - a}}{\sum_{i=1}^N e^{x_i - a}}$$

often chosen $a = \max_i x_i$. Show that the above identity holds.

For some arbitrary constant C , we have: $\hat{C} e^{x_i} = e^{\log(C) x_i} e^{x_i + \log(C)}$

$$\frac{e^{x_i}}{\sum_{i=1}^N e^{x_i}} = \frac{C e^{x_i}}{C \sum_{i=1}^N e^{x_i}} = \frac{e^{x_i + \log(C)}}{\sum_{i=1}^N e^{x_i + \log(C)}}$$

Since C is arbitrary, we can set $\log(C) = -a$ and get $\frac{e^{x_i - a}}{\sum_{i=1}^N e^{x_i - a}}$