HW9

The "softmax" function takes a list $[x_1, \ldots, x_n]$ of real numbers and an index $i \in \{1, \ldots, n\}$:

$$\text{softmax}([x_1, \ldots, x_n], i) = \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_j}}$$

It's called softmax because it usually turns the largest element of the list to something close to 1, and the others to something close to zero

e.g. $\text{softmax}([-4, 1.5, 5], 1) = 0.00012$

$\text{softmax}([-4, 1.5, 5], 2) = 0.029$

$\text{softmax}([-4, 1.5, 5], 3) = 0.971$

(a) Show for any real number b:

$$\text{softmax}([x_1 + b, \ldots, x_n + b], i) = \text{softmax}([x_1, \ldots, x_n], i)$$

(b) Express the sigmoid function $\sigma(x)$ in terms of softmax:

$$\sigma(x) = \text{softmax}(\boxed{}, \boxed{})$$

(c) Suppose we try to compute $\sigma(-740)$ directly, using Python. What happens?

(d) How can we exploit result (a) to compute $\sigma(-740)$? Show an implementation of $\sigma(x)$:

```
def stabler_sigmoid(x):
```