

Assignment 1

Exercises on Numpy

You have access to the NumPy python library as `np` for the whole assignment

Randomization

Write a function called `randomization` that takes as input a positive integer `n`, and returns `A`, a random `n x 1` Numpy array.

Operations

Write a function called `operations` that takes as input two positive integers `h` and `w`, makes two random matrices `A` and `B`, of size `h x w`, and returns `A`, `B`, and `s`, the sum of `A` and `B`.

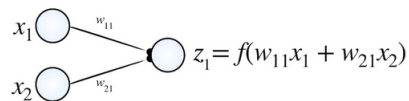
Norm

Write a function called `norm` that takes as input two Numpy column arrays `A` and `B`, adds them, and returns `s`, the L2 norm of their sum.

Neural Network

write a function `neural_network`, which will apply a neural network operation with 2 inputs and 1 output and a given weight matrix.

The output, z_1 , is given by $z_1 = f(w_{11}x_1 + w_{21}x_2)$:



where f is a specified nonlinear function, and it is usually the hyperbolic tangent function, \tanh .

If we express our inputs and weights as matrices, as shown here,

$$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad w = \begin{bmatrix} w_{11} \\ w_{21} \end{bmatrix}$$

then we can develop an elegant mathematical expression: $z_1 = \tanh(w^T \vec{x})$.

Scalar function

Start with writing a scalar function `scalar_function`, which will apply the following operation with input `x` and `y`.

$$f(x, y) = \begin{cases} x \cdot y, & \text{if } x \leq y \\ x/y, & \text{else.} \end{cases}$$

Note that `x` and `y` are scalars.

Vector function

`scalar_function` can only handle scalar input, we could use the function `np.vectorize()` turn it into a vectorized function. Note that the input argument of `np.vectorize()` should be a scalar function, and the output of `np.vectorize()` is a new function that can handle vector input.

Please write a vector function `vector_function`, which will apply the operation $f(x,y)$ defined above element-wisely with input vectors with same dimension `x` and `y`.

NOTE: use the function `scalar_function` in the previous exercise.