For these exercises, you should review the notes on Neural Networks.

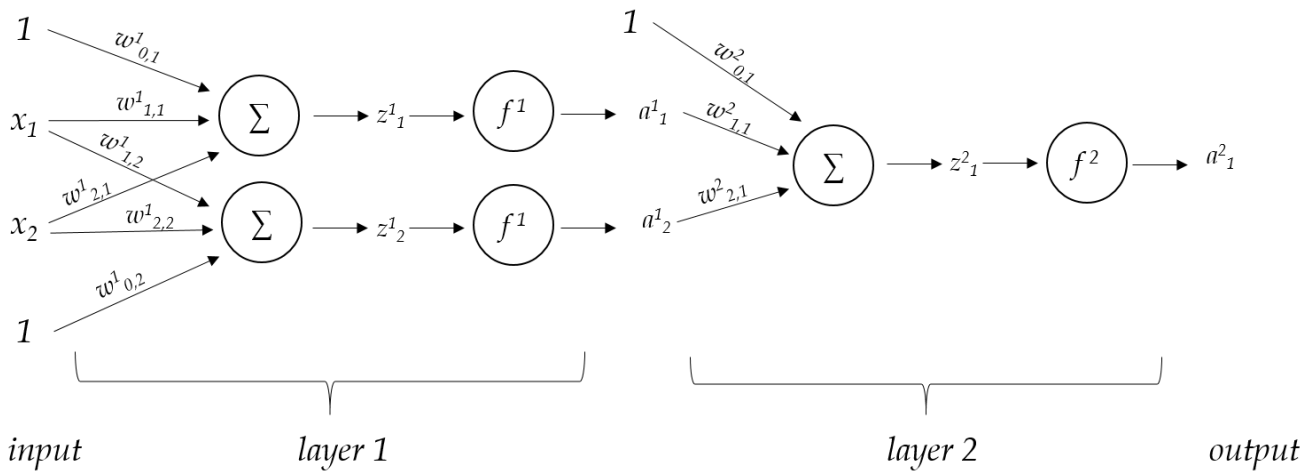
1) Prediction

Consider the following data set:

```
X = np.array([[0, 1, 2],
              [0, 1, 2]])
Y = np.array([[0, 1, 0]])
```

The columns of X and Y are the data points and corresponding labels.

We will be looking at the behavior of the following simple two-layer network:



Assume that within each layer, each unit has the step activation function $f(z)$ given by

$$f(z) = \begin{cases} 1 & \text{if } z > 0 \\ 0 & \text{otherwise} \end{cases}.$$

Let the weights in the first layer (layer 1) be:

- $w^1_{0,1} = -0.5, w^1_{1,1} = 1, w^1_{2,1} = 0$
- $w^1_{0,2} = 1.5, w^1_{1,2} = -1, w^1_{2,2} = 0$

1A) Enter a matrix Z where each column represents the outputs of the hidden units ($f^1(z^1_1)$ and $f^1(z^1_2)$) for each of the input vectors in `X`.

Enter a Python list of lists [[a,b,c],[d,e,f]], each list is a row of the matrix.

[[0, 1, 1], [1, 1, 0]]

Submit View Answer **100.00%**

You have infinitely many submissions remaining.

1B) Pick weights for the second layer $w^2_{0,1}, w^2_{1,1}, w^2_{2,1}$ so that the desired outputs are predicted correctly.

Enter a Python list of 3 numbers $[w^2_{0,1}, w^2_{1,1}, w^2_{2,1}]$ [-1.5, 1, 1]

Submit View Answer **100.00%**

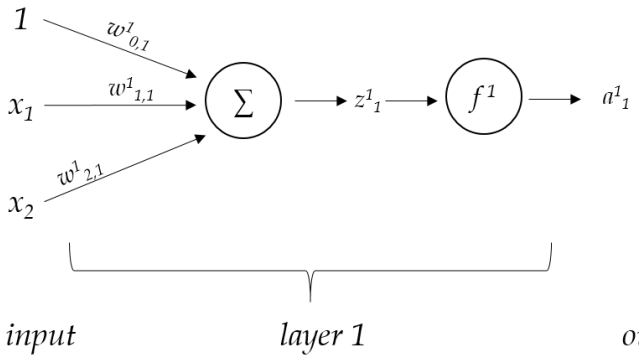
You have infinitely many submissions remaining.

2) Training

Now, we will consider the classification of a different set of `X` and `Y` data, with a different single layer network having the following structure and activation function:

$$z = w^1_{1,1}x_1 + w^1_{2,1}x_2 + w^1_{0,1}$$

In this network we have $f^1(z) = z$, so our output $a^1_1 = z^1_1$.



Assume the initial weights are $w^1_{0,1} = 1, w^1_{1,1} = 1, w^1_{2,1} = 1$, and the step size is 0.5 (not usually a good idea, but okay for now).

The current training example is $x^{(i)} = [1, 2]^T$, $y^{(i)} = -1$.

2A) What is the output value a^1_1, given current input $x^{(i)}$ and the current weights?

Enter a number 4

Submit View Answer **100.00%**

You have infinitely many submissions remaining.

2B) What will the values of weights $w^1_{0,1}, w^1_{1,1}, w^1_{2,1}$ be after one step of stochastic gradient descent at the given training example $x^{(i)} = [1, 2]^T$, $y^{(i)} = -1$ using our definition of hinge loss $L_h(v) = max(0, 1 - v)$?

Enter a Python list of 3 numbers $[w^1_{0,1}, w^1_{1,1}, w^1_{2,1}]$ (to 3 decimal places)

[0.5, 0.5, 0]

Submit View Answer **100.00%**

You have infinitely many submissions remaining.

2C) What would the output value a^1_1 be, for this same input x, with these new weights?

Enter a number 1

Submit View Answer **100.00%**

You have infinitely many submissions remaining.

2D) What would happen to the v_i if we did another SGD update, for that same point, with step size 0.5, as before?

Enter a Python list of 3 numbers $[w_{0,1}^1, w_{1,1}^1, w_{2,1}^1]$ [0, 0, -1]

Submit View Answer **100.00%**

You have infinitely many submissions remaining.

2E) Now what would the output be?

Enter a number -2

Submit View Answer **100.00%**

You have infinitely many submissions remaining.

2F) What if we do one more update, for that same point?

Enter a Python list of 3 numbers $[w_{0,1}^1, w_{1,1}^1, w_{2,1}^1]$ [0, 0, -1]

100.00%

You have infinitely many submissions remaining.

Solution: `[0.0, 0.0, -1.0]`

Explanation:

Nothing happens; the derivatives are zero since the hinge loss is zero.