- Vectorisation does <u>not</u> allow you to compute forward prop. i
L-layer NN w/out an explicit for-loop (or any other expli
iterative loop) over layers l=1,2,...,L. We can't avoid the for
iteration over the computations among layers

? - layer_dims = [n_x, 4, 3, 2, 1]; how to initialise model parameter

```
for (i in range(1, len(layer_dims))):
    parameter['w'+str(i)] = np.random.randn(layers[i] layers[i ))*
    parameter['b'+str(i)] = np.random.randn(layers[i],1)*0
```

- During for.prop., in the forward function for a layer l, we
to know what the act. func. in a layer is.
During backprop, the corresp. backward func. also needs to
what the act. func for layer l is, since the gradient depe
on it

<u>Week 4 Programming Assignment: Building your Deep NN:</u> <u>by Step</u>

- Objectives:
  - Develop an intuition of the overall structure of a NN
  - Write functions that would help us decompose our code and the process of building a NN (e.g. for.prop., backprop., log los
  - Initialise/update parameters

- Contents:
  1. Packages
  2. Outline
     - Init. params. for a two-layer net. and for an L-layer net.
     - Implement the for. prop. module
     - Compute loss
     - Implement the backprop. module
     - Update parameters